Cement Library V1.1

User Manual Level 2

07/2011

EIO0000000886.00

www.schneider-electric.com

Schneider Electric

The information provided in this documentation contains general descriptions and/or technical characteristics of the performance of the products contained herein. This documentation is not intended as a substitute for and is not to be used for determining suitability or reliability of these products for specific user applications. It is the duty of any such user or integrator to perform the appropriate and complete risk analysis, evaluation and testing of the products with respect to the relevant specific application or use thereof. Neither Schneider Electric nor any of its affiliates or subsidiaries shall be responsible or liable for misuse of the information contained herein. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us.

No part of this document may be reproduced in any form or by any means, electronic or mechanical, including photocopying, without express written permission of Schneider Electric.

All pertinent state, regional, and local safety regulations must be observed when installing and using this product. For reasons of safety and to help ensure compliance with documented system data, only the manufacturer should perform repairs to components.

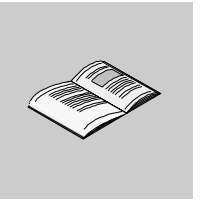
When devices are used for applications with technical safety requirements, the relevant instructions must be followed.

Failure to use Schneider Electric software or approved software with our hardware products may result in injury, harm, or improper operating results.

Failure to observe this information can result in injury or equipment damage.

© 2011 Schneider Electric. All rights reserved.

Table of Contents



Safety Information **9**
About the Book **15**
Part I General Information **17**
Chapter 1 Basics of Cement Library **19**
Basics of Cement Library 19
Chapter 2 Functional Description **25**
Functional Description 25
Chapter 3 Function Blocks **37**
Function Blocks 37
Chapter 4 Representation in the HMI **41**
Cement Library Template 42
HMI Object List 44
Popup Representation 47
Commonly Used Tabs 51
Chapter 5 Using the Cement Library **55**
Operator Logging 56
Alarm Logging and Alarm Summary 57
Dim/Unavailable Buttons 58
Alarms in the Popup 59
Access Level Support 60
Enable Use of Double Monitor 62
Acknowledge Alarms 65
Local Language Support 66
Navigation to Trend and Alarm Window from the Popup 68
Trend Assignment 69
Color Customization 70
Navigation Buttons 73
Information Popup 74
Out of Service Mode 75
Chapter 6 Implementation Paradigms **77**
Defining Startup Function in Vijeo Citect 78
Scaling of Values for Visualization 88
Eng. Min/Eng. Max Property of the Control Module 89

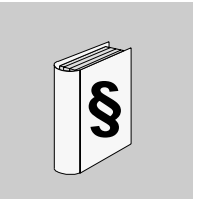
Chapter 7 Instantiation in UAG 91
UAG Customization 92
UAG Generator 94
Part II Object Description 103
Chapter 8 Equipment Control 105
8.1 Mot1d: Motor 1 Direction 106
Mot1d - Description 107
Mot1d - Representation as SCoD in UAG 119
Mot1d - Representation in the HMI 122
Mot1d_St Derived Data Type 130
8.2 Mot2d: Motor 2 Direction 133
Mot2d - Description 134
Mot2d - Representation as SCoD in UAG 147
Mot2D - Representation in the HMI 150
Mot2d_St Derived Data Type 158
8.3 AnlgIn: Analog Input 161
AnlgIn - Description 162
AnlgIn - Representation as SCoD in UAG 167
AnlgIn - Representation in the HMI 169
AnlgIn_St Derived Data Type 172
8.4 AnlgOut: Analog Output 174
AnlgOut - Description 175
AnlgOut - Representation as SCoD in UAG 179
AnlgOut - Representation in the HMI 181
AnlgOut_St Derived Data Type 184
8.5 DigIn: Digital Input 185
DigIn - Description 186
DigIn - Representation as SCoD in UAG 190
DigIn - Representation in the HMI 192
DigIn_St Derived Data Type 196
8.6 DigOut: Digital Output 198
DigOut - Description 199
DigOut - Representation as SCoD in UAG 202
DigOut - Representation in the HMI 203
DigOut_St Derived Data Type 206
8.7 Mot1dVS: Motor 1 Direction Variable Speed 207
Mot1dVS - Description 208
Mot1dVS - Representation as SCoD in UAG 220
Mot1dVS - Representation in the HMI 223
Mot1dVS_St Derived Data Type 233

8.8 DigAct: Digital Actuator 236
DigAct - Description 237
DigAct - Representation as SCoD in UAG 246
DigAct - Representation in the HMI 249
DigAct_St Derived Data Type 257
8.9 MotAct: Motor Actuator 260
MotAct - Description 261
MotAct - Representation as SCoD in UAG 272
MotAct - Representation in the HMI 275
MotAct_St Derived Data Type 284
8.10 AnlgAct: Analog Actuator 287
AnlgAct - Description 288
AnlgAct - Representation as SCoD in UAG 300
AnlgAct - Representation in the HMI 303
AnlgAct_St Derived Data Type 313
8.11 MotBasic: Motor Basic 316
MotBasic - Description 317
MotBasic - Representation as SCoD in UAG 324
MotBasic - Representation in the HMI 326
MotBasic_St Derived Data Type 332
Chapter 9 MMM Process 335
9.1 GrpCtrl: Group Control. 336
GrpCtrl - Description 337
GrpCtrl - Representation as SCoD in UAG 344
GrpCtrl - Representation in the HMI 347
GrpCtrl_St Derived Data Type 355
9.2 GrpStrtMgt: Group Start Management 357
GrpStrtMgt - Description 358
GrpStrtMgt - Representation as SCoD in UAG 362
GrpStrtMgt - Representation in the HMI 366
GrpStrtMgt_St Derived Data Type 370
9.3 GrpStpMgt: Group Stop Management 371
GrpStpMgt - Description 372
GrpStpMgt - Representation as SCoD in UAG 376
GrpStpMgt - Representation in the HMI 380
GrpStpMgt_St Derived Data Type 384
9.4 BCnvyr1d: Belt Conveyor 1 Direction 385
BCnvyr1d - Description 386
BCnvyr1d - Representation as SCoD in UAG 400
BCnvyr1d - Representation in the HMI 403
BCnvyr1d_St Derived Data Type 414

9.5 RouteMgt: Route Management 418
RouteMgt - Description 419
RouteMgt - Representation as SCoD in UAG 427
RouteMgt - Representation in the HMI 430
RouteMgt_St Derived Data Type 436
9.6 BagFltr: Bag Filter Sequence Block 439
BagFltr - Description 440
BagFltr - Representation as SCoD in UAG 448
BagFltr - Representation in the HMI 451
BagFltr_St Derived Data Type 458
Chapter 10 Cement Process 461
10.1 GasAnalyser: Gas Analyser 462
GasAnalyser - Description 463
GasAnalyser - Representation as SCoD in UAG 473
GasAnalyser - Representation in the HMI 476
GasAnalyser_St Derived Data Type 484
10.2 MultiFuelCtrl: Multi Fuel Control 487
MultiFuelCtrl - Description 488
MultiFuelCtrl - Representation as SCoD in UAG 497
MultiFuelCtrl - Representation in the HMI 503
MFuelCtrl_St Derived Data Type 509
10.3 GirthGearLubCtrl: Girth Gear Lubrication Control 512
GirthGearLubCtrl - Description 513
GirthGearLubCtrl - Representation as SCoD in UAG 519
GirthGearLubCtrl - Representation in the HMI 522
GirthGearLubCtrl_St Derived Data Type 528
Chapter 11 Generic Process 531
11.1 SPBool: Set Point Bool 533
SPBool - Description 534
SPBool - Representation as SCoD in UAG 536
SPBool - Representation in the HMI 537
SPBool_St Derived Data Type 538
11.2 SPInt: Set Point Integer 539
SPInt - Description 540
SPInt - Representation as SCoD in UAG 542
SPInt - Representation in the HMI 543
SPInt_St Derived Data Type 546
11.3 SPReal: Set Point Real 547
SPReal - Description 548
SPReal - Representation as SCoD in UAG 550
SPReal - Representation in the HMI 551
SPReal_St Derived Data Type 554

11.4 RatioCtrl: Ratio Controller 555
RatioCtrl - Description 556
RatioCtrl - Representation as SCoD in UAG 563
RatioCtrl - Representation in the HMI 566
RatioCtrl_St Derived Data Type 572
11.5 Totaliser: Totaliser 575
Totaliser - Description 576
Totaliser - Representation as SCoD in UAG 578
Totaliser - Representation in the HMI 579
Totaliser_St Derived Data Type 582
11.6 EgySchE: Energy Scheduler Event Based 583
EgySchE - Description 584
EgySchE - Representation as SCoD in UAG 586
EgySchE - Representation in the HMI 587
EgySchE_St Derived Data Type 592
11.7 EgySchT: Energy Scheduler Time Based 594
EgySchT - Description 595
EgySchT - Representation as SCoD in UAG 597
EgySchT - Representation in the HMI 598
EgySchT_St Derived Data Type 603
11.8 ILCK: Interlock 605
ILCK - Description 606
ILCK - Representation as SCoD in UAG 608
ILCK_St Derived Data Type 609
11.9 PIDB: Proportional, Integral and Derivative Controller Block 610
PIDB - Description 611
PIDB - Representation as SCoD in UAG 619
PIDB - Representation in the HMI. 621
PIDB_St Derived Data Type 629
Glossary 633
Index 643

Safety Information



Important Information

NOTICE

Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.

The addition of this symbol to a Danger or Warning safety label indicates that an electrical hazard exists, which will result in personal injury if the instructions are not followed.

This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ DANGER

DANGER indicates an imminently hazardous situation which, if not avoided, **will result in** death or serious injury.

⚠ WARNING

WARNING indicates a potentially hazardous situation which, if not avoided, **can result in** death or serious injury.

⚠ CAUTION

CAUTION indicates a potentially hazardous situation which, if not avoided, **can result in** minor or moderate injury.

CAUTION

CAUTION, used without the safety alert symbol, indicates a potentially hazardous situation which, if not avoided, **can result in** equipment damage.

PLEASE NOTE

Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction and operation of electrical equipment and its installation, and has received safety training to recognize and avoid the hazards involved.

BEFORE YOU BEGIN

Do not use this product on machinery lacking effective point-of-operation guarding. Lack of effective point-of-operation guarding on a machine can result in serious injury to the operator of that machine.

⚠ WARNING

UNGUARDED MACHINERY CAN CAUSE SERIOUS INJURY

- Do not use this software and related automation equipment on equipment which does not have point-of-operation protection.
- Do not reach into machinery during operation.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

This automation equipment and related software is used to control a variety of industrial processes. The type or model of automation equipment suitable for each application will vary depending on factors such as the control function required, degree of protection required, production methods, unusual conditions, government regulations, etc. In some applications, more than one processor may be required, as when backup redundancy is needed.

Only the user can be aware of all the conditions and factors present during setup, operation, and maintenance of the machine; therefore, only the user can determine the automation equipment and the related safeties and interlocks which can be properly used. When selecting automation and control equipment and related software for a particular application, the user should refer to the applicable local and national standards and regulations. The National Safety Council's Accident Prevention Manual (nationally recognized in the United States of America) also provides much useful information.

In some applications, such as packaging machinery, additional operator protection such as point-of-operation guarding must be provided. This is necessary if the operator's hands and other parts of the body are free to enter the pinch points or other hazardous areas and serious injury can occur. Software products alone cannot protect an operator from injury. For this reason the software cannot be substituted for or take the place of point-of-operation protection.

Ensure that appropriate safeties and mechanical/electrical interlocks related to point-of-operation protection have been installed and are operational before placing the equipment into service. All interlocks and safeties related to point-of-operation protection must be coordinated with the related automation equipment and software programming.

NOTE: Coordination of safeties and mechanical/electrical interlocks for point-of-operation protection is outside the scope of the Function Block Library, System User Guide, or other implementation referenced in this documentation.

START-UP AND TEST

Before using electrical control and automation equipment for regular operation after installation, the system should be given a start-up test by qualified personnel to verify correct operation of the equipment. It is important that arrangements for such a check be made and that enough time is allowed to perform complete and satisfactory testing.

⚠ CAUTION

EQUIPMENT OPERATION HAZARD

- Verify that all installation and set up procedures have been completed.
- Before operational tests are performed, remove all blocks or other temporary holding means used for shipment from all component devices.
- Remove tools, meters, and debris from equipment.

Failure to follow these instructions can result in injury or equipment damage.

Follow all start-up tests recommended in the equipment documentation. Store all equipment documentation for future references.

Software testing must be done in both simulated and real environments.

Verify that the completed system is free from all short circuits and grounds, except those grounds installed according to local regulations (according to the National Electrical Code in the U.S.A, for instance). If high-potential voltage testing is necessary, follow recommendations in equipment documentation to prevent accidental equipment damage.

Before energizing equipment:

- Remove tools, meters, and debris from equipment.
- Close the equipment enclosure door.
- Remove ground from incoming power lines.
- Perform all start-up tests recommended by the manufacturer.

OPERATION AND ADJUSTMENTS

The following precautions are from the NEMA Standards Publication ICS 7.1-1995 (English version prevails):

- Regardless of the care exercised in the design and manufacture of equipment or in the selection and ratings of components, there are hazards that can be encountered if such equipment is improperly operated.
- It is sometimes possible to misadjust the equipment and thus produce unsatisfactory or unsafe operation. Always use the manufacturer's instructions as a guide for functional adjustments. Personnel who have access to these adjustments should be familiar with the equipment manufacturer's instructions and the machinery used with the electrical equipment.
- Only those operational adjustments actually required by the operator should be accessible to the operator. Access to other controls should be restricted to prevent unauthorized changes in operating characteristics.

⚠ WARNING

WRONG CONFIGURATION/ENGINEERING

- Any application developed with the help of Cement Library V1.1 by system integrator or customer needs to be thoroughly tested by them.
- Cement Library V1.1 is not responsible for any data loss during the power failure of any system, that is, PLC, SCADA, and so on.
- Schneider Electric is not responsible for any wrong application design developed by system integrator or customer.
- Schneider Electric is not responsible for any modifications done by the user in the delivered products.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

⚠ WARNING

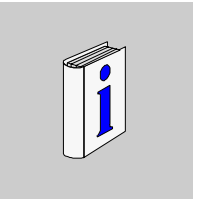
OPERATION OF THE PLANT BY UNAUTHORIZED PEOPLE

Before using or adopting the access levels for use in specific application the engineer must:

- conduct a safety analysis for the application and equipment installed
- verify that all access levels are appropriate for the equipment or function in the installation
- thoroughly test all functions during verification and commissioning
- provide independent paths for critical functions (emergency stop, overlimit conditions, and so on) according to the safety analysis and applicable codes and regulations.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

About the Book



At a Glance

Document Scope

This document provides information on understanding and operating a project developed with Unity Application Generator (UAG) and the Cement Library V1.1.

It is not intended to cover any development procedures and internal functionality details.

The Cement Library is intended for end users and system integrators.

Users of the Cement Library must have very good knowledge in the employment of:
- Unity Application Generator V3.3
- Unity Pro XL V4.1 and V5.0
- Vijeo Citect V7.1
- OPC Factory Server V3.3

Validity Note

This manual is valid for the listed versions of the following products:
- Unity Application Generator V3.3
- Unity Pro XL V4.1 and V5.0
- Vijeo Citect V7.1
- OPC Factory Server V3.3

Related Documents

Title of Documentation	Reference Number
Unity Application Generator Basic User Manual	33002830
Unity Application Generator Extended User Manual	33003669
Unity Application Generator SCoD Editor Manual	33002608
Unity Application Generator Quick Start Guide	33003222

You can download these technical publications and other technical information from our website at www.schneider-electric.com.

Product Related Information

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

Only persons with expertise in the design and programming of control systems should be allowed to program, install, alter, and apply this product.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

Before being used in an industrial application, examples must be suitably adapted to the specific functions and safety requirements of the application concerned. All examples in this manual are given for information only.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

User Comments

We welcome your comments about this document. You can reach us by e-mail at techcomm@schneider-electric.com.

General Information

I

Introduction

This part provides general information on the Cement Library for Vijeo Citect V7.1.

What's in this Part?

This part contains the following chapters:

Chapter	Chapter Name	Page
1	Basics of Cement Library	19
2	Functional Description	25
3	Function Blocks	37
4	Representation in the HMI	41
5	Using the Cement Library	55
6	Implementation Paradigms	77
7	Instantiation in UAG	91

Basics of Cement Library

1

Basics of Cement Library

General

The Cement Library is an open software engineering tool based on Unity PLCs, Unity Pro, UAG, and Vijeo Citect SCADA.

A set of standardized objects for SCADA (symbols) and PLC (DFBs) provides a rapid application development. All the information related to the objects (actuators, sensors, measurements, loops, PLCs, and so on) are stored in a single data structure.

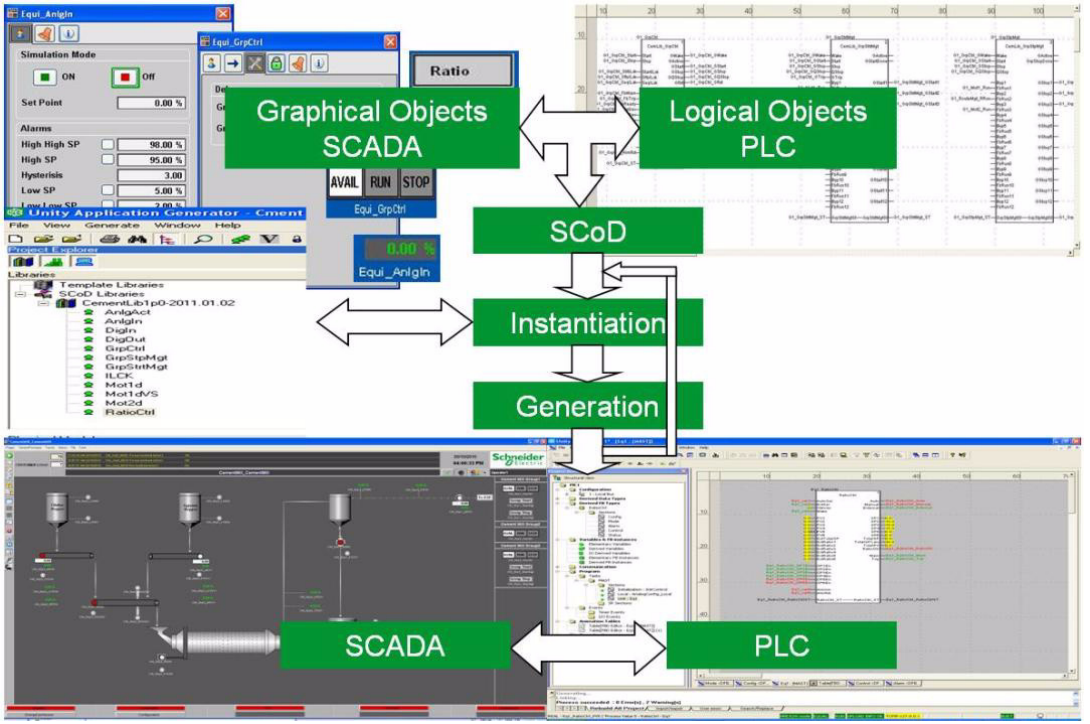
Due to an open interface, extensions and modifications are easily done. This enables you to adapt certain functions for specific needs.

Same Name for All Objects

Each object consists of a SCoD, a Unity DFB, and a Vijeo Citect Genie. The names of all of these object elements are same, for instance the function block `GrpCtrl`.

Interaction of the Cement Library

The following figure shows the interaction of the Cement Library with UAG Application during the development of an application.



Families and DFBs of the Cement Library

The Cement Library has a set of families in the Unity Pro, each consisting of a number of Derived Function Blocks (DFB). The installation setup program installs the library comprising of the families which consist of the Derived Function Blocks (DFBs) in the Unity Pro Libset.

The following table provides the details of DFB name and description for each set of Cement Library family in the Unity Pro:

Unity Pro		
Family	**DFB Name**	**Comments**
Equipment Control	Mot1D	Motor 1 Direction
	Mot2D	Motor 2 Direction
	AnlgAct	Analog Actuator
	DigAct	Digital Actuator
	MotAct	Motor Actuator
	Mot1DVS	Motor 1 Direction Variable Speed
	MotBasic	Motor Basic
	DigIn	Digital Input
	DigOut	Digital Output
	AnlgIn	Analog Input
	AnlgOut	Analog Output
MMM Process (Mining, Minerals and Metals)	GrpCtrl	Group Control
	GrpStrtMgt	Group Start Management
	GrpStpMgt	Group Stop Management
	RouteMgt	Route Management
	BagFltr	Bag Filter Sequence
	BCnvyr1d	Belt Conveyor 1 Direction
Cement Process	GirthGearLubCtrl	Girth Gear Lubrication Control
	GasAnalyser	Gas Analyser

Unity Pro		
Family	**DFB Name**	**Comments**
Devices	CATV31M	CANopen ATV31M
	CATV61M	CANopen ATV61M
	CATV71M	CANopen ATV71M
	CTesysT	CANopen TesysT
	EATV71	Ethernet ATV71
	MBATV31M	Modbus ATV31
	MBATV61	Modbus ATV61
	MBATV71	Modbus ATV71
	MBEM6300	Modbus Energy Meter6300
	MBEM7300	Modbus Energy Meter7300
	MBPM710	Modbus Power Meter710
	MBPM800	Modbus Power Meter800
	MBTesysT	Modbus TeSys T
Generic Process	PIDB	Proportional, Integral, and Derivative Controller Block
	Ilck	Interlock
	Totaliser	Totaliser
	EgySchE	Energy Scheduler Event Based
	EgySchT	Energy Scheduler Time Based
	RatioCtrl	Ratio Controller
	SPBool	Set Point Bool
	SPInt	Set Point Integer
	SPReal	Set Point Real

Unity Pro		
Family	**DFB Name**	**Comments**
Communication	ASRTPBdg	CANopen Gateway for Advantys Devices
	CClient	CANopen Client
	CPortM	CANopen Port M340
	CPortP	CANopen Port Premium
	CScanner	CANopen Scanner
	EClient	Ethernet Client
	EGtwMB	Ethernet Gateway Modbus
	EPortM	Ethernet Port M340
	EPortP	Ethernet Port Premium
	EPortQ	Ethernet Port Quantum
	EScanner	Ethernet Scanner
	MBClientBs	Modbus Client Basic
	MBPortM	Modbus Port M340
	MBPortP	Modbus Port Premium
	MBScanner	Modbus Scanner
	Statistics	Statistics

Functional Description

2

Functional Description

Overview

The Cement Library provides the basic and advanced functions to be used with Unity Application Generator (UAG) in terms of process automation.

The Unity Pro Libset includes the following 6 functional families:

- **Equipment Control**
- **Generic Process**
- **MMM Process**
- **Cement Process**
- **Devices**
- **Communication**

Each single function realizes the following parts:

- a derived function block for Unity Pro (DFB)
- a HMI Genie and popup for Vijeo Citect
- a SCoD in UAG

Both PLC and HMI parts are bundled by the UAG which is realized as Smart Control Device (SCoD). All the functions are based on a common feature that enables easy handling of the library. Thus, all the functions share the same operating mode, behavior and alarm handling with an identical communication interface between the PLC and the HMI.

Equipment Control

The family **Equipment Control** is comprised of elements that are used to control different equipments like motors, valves and actuators, and also process PLC input and output signals of analog and digital types.

The following table describes the functions equipped according to the library:

Signal Type	Single Functions
Mot1D	• controls electrical motors with one direction of rotation and constant speed • controls the motor from different sources, according to the chosen operation mode (Auto, Manual and Local) • switches to Out of Service mode, when the motor is under maintenance mode • supervises motor inoperable conditions • supervises run feedback • supervises process feedback (can be bypassed) • stops (conditional) on a detected error • supervises the operable or inoperable status of all interlocks • generates alarms and alerts • displays the value and status in the HMI • example of applications: motor, pump, blower, conveyor, and so on
Mot2D	• controls electrical motors with two direction of rotation and constant speed • controls the motor from different sources, according to the chosen operation mode (Auto, Manual and Local) • switches to Out of Service mode, when the motor is under maintenance mode • supervises motor inoperable conditions • supervises reverse run feedback • supervises forward run feedback • supervises process feedback (can be bypassed) • stops (conditional) on a detected error • supervises the operable or inoperable status of all interlocks • generates alarms and alerts • displays the value and status in the HMI • example of applications: motor, pump, blower, conveyor, and so on
Mot1DVS	• controls electrical motors with one direction of rotation and variable speed • controls the motor from different sources, according to the chosen operation mode (Auto, Manual and Local) • switches to Out of Service mode, when the motor is under maintenance mode • supervises motor inoperbale conditions • stops (conditional) on a detected error • supervises motor run feedback • supervises motor speed (speed of rotation) • supervises process feedback (can be bypassed) • supervises the operable or inoperable status of all interlocks in the popup • displays the value and status in the HMI • generates alarms and alerts • example of applications: motor, pump, blower, conveyor, and so on

Signal Type	Single Functions
MotBasic	• controls electrical motors with one direction of rotation • stops (conditional) on a detected error • supports only Auto (sequential) mode of operation • supervises run feedback • supervises the operable or inoperable status of all interlocks • displays the value and status in the HMI • generates alarms and alerts • example of applications: motor, pump, blower, conveyor, and so on
AnlgIn	• scales the raw input value from analog input card based on the scale range specified • replaces the input signal with pre-defined value when a hardware error is detected (optional) • supervises high and low level alarm limits • manually simulates the input signal from the HMI system • displays the value and status in the HMI
AnlgOut	• de-scales the set point value to analog output card as per the hardware used or based on the scale range specified • restricts output within pre-defined high and low output limits • replaces the input signal with pre-defined value when any hardware error is detected (optional) • manually replaces the output signal from the HMI system • displays the value and status in the HMI
DigIn	• evaluates the digital input signal from the digital input card • supervises the alarm and normal state • replaces the input signal with pre-defined value when any hardware error is detected (optional) • enables the selection of HMI symbols for display of genies with letters A...Z • replaces the input signal in simulation with simulation value from the HMI system • displays the value and status in the HMI
DigOut	• evaluates the digital output signal to the digital output card • replaces the output signal in Manual mode from the HMI system • replaces the input signal with pre-defined value when any hardware error is detected (optional) • displays the value and status in the HMI

Signal Type	Single Functions
DigAct	• controls pneumatic/electrical/electro-pneumatic actuators without motor and with defined positions used for driving valves, gates, dampers, and so on • supports Auto and Manual operating modes • switches to Out of Service mode, when the motorized actuator is under maintenance mode • supervises position feedback • displays the operable or inoperable status of all interlocks in the popup • displays the number of active alarms in the popup • displays the value and status in the HMI • enables the selection of HMI symbols for display (example, valves, dampers, divertors, gates, and so on)
MotAct	• controls electrical actuators and motor with defined positions used for driving valves, gates, dampers, and so on • supports Auto, Manual and Local operating modes • switches to Out of Service mode, when the motorized actuator is under maintenance mode • trips the actuator and generates alarm if the actuator is inoperable • supervises the run feedback • supervises the position feedback • displays the operable or inoperable status of all interlocks in the popup • enables the selection of HMI symbols for display (for example, valves, dampers, divertors, gates, and so on)
AnlgAct	• controls the position of equipment by a two-way driven actuator • supports Auto, Manual and Local operating modes • switches to Out of Service mode, when the analog actuator is under maintenance mode • supervises actuator run feedback • supervises actuator position feedback • displays the operable or inoperable status of all interlocks in the popup

Generic Process

The group **Generic Process** is used to control the ratio of ingredients, calculate the control variable value based on the deviation derived and provide information on ratio controller, set points of bool, real, integer, and so on.

The following table describes the functions equipped according to the library:

Signal Type	Single Functions
PIDB	• supports Auto, Manual, Cascade and External modes • goes to default mode during initialization • tracks Auto mode set point, Manaul mode set point and process value (optional) • calculates the manipulated variable value based on the deviation derived • generates alarms and alerts • displays the value and status in the HMI
Ilck	• processes different interlock conditions into single interlock output that shall be used by motors, actuators, and so on • supports 8 interlock conditions • displays the value and status in the HMI
Totaliser	• initializes the **Totalising** command as input to the block • displays the totalized value • starts totalising the input value given as an input pin • displays the value and status in the HMI
EgySchE	• schedules the application action based events • supports the Bool type and Real type output • displays the remaining time, scheduler on/off status • generates alarms if the schedule is cancelled • displays the value and status in the HMI
EgySchT	• schedules the output as per time set from HMI • supports the Bool type output • displays the remaining time, scheduler on/off status • generates alarms if the schedule is cancelled • displays the value and status in the HMI
RatioCtrl	• controls the ratio of ingredients of a continous process • supports Auto, Manual and External modes • calculates the percentage set point of each parameter • generates an alarm if the input and output set points are not equal • supports upto 6 outputs • displays the value and status in the HMI
SPBool	• accepts the input set point in bool format • allows the operator to enter the set points of bool format from HMI • configures the input type (latch/unlatch/pulse) during instantiation of the block • displays the value and status in the HMI

Signal Type	Single Functions
SPInt	● accepts the input set point in integer format ● allows the operator to enter the set points of INT type from HMI ● configures the output HIGH and output LOW values during instantiation of the block ● enables the user to configure the delay in changing in set point value from old value to new value through HMI ● displays the value and status in the HMI
SPReal	● accepts the input set point in real format ● allows the operator to enter the set points of Real type from HMI ● configures the output HIGH and output LOW values during instantiation of the block ● enables the user to configure the delay in changing in set point value from old value to new value through HMI ● displays the value and status in the HMI

MMM Process

The family **MMM Process** comprises of elements that are used to control various applications.

This family is used to control start/stop of the device, and manage sequential start/stop of the devices in the group, devices routes (motors/actuators), bag filter purging operation, and so on.

The following table describes the functions equipped according to the library:

Signal Type	Single Functions
GrpCtrl	● controls the start/stop of the devices in the group ● checks the availability of the group in all the devices ● turn onto Sleep mode, when the group is not active ● Initiates Group stop **GStop** command if there is an inoperability of sequence interlock ● initiates Group quick stop **GQStop** command if there is an inoperability of safety interlock ● resets the alarms whenever a **Reset** command is issued ● displays the value and status in the HMI
GrpStrtMgt	● manages the sequential start of the devices in the group ● plays an intermediate role between group control and device (motors/actuators) blocks ● introduces delay between starting of the devices in the group ● holds the starting of devices in the group (optional) ● displays the value and status in the HMI

Signal Type	Single Functions
GrpStpMgt	• manages the sequential stopping of the devices in the group • plays an intermediate role between group control and device (motors/actuators) blocks • introduces delay between stopping of the devices in the group • holds the stopping of devices in the group (optional) • displays the value and status in the HMI
RouteMgt	• manages the start and stop of sub-groups comprising of functional, elementary and composite blocks belonging to the same group • supports upto 5 routes • supports the operable and inoperable of all interlocks in the HMI • monitors all devices, alerts, trip, run feedback of the essential and all stop devices continuously • displays the detailed active alarms in the alarm tab of the popup • displays the value and status in the HMI
BagFltr	• supervises and controls the bag filter purging operation • supports only Auto mode of operation • supports 18 purging outputs (solenoid values) • displays the interlock status (operable or inoperable) in the popup • displays the number of active alarms in the popup of the motor • displays the value and status in the HMI
BeltCnvyr1d	• supports Auto, Manual and Local operating modes • supervises run feedback • supervises process feedback (can be bypassed) • supervises specific or unique feedbacks of BeltCnvyr1d (BSS, PCS, and so on) • initiates **Run** and **Stop** commands from the popup • monitors **Pull Cord** switch feedback, belt tension feedback, belt chute and empty run of conveyor feedback continuously • displays input values of load and energy in the popup • generates alarms and alerts • displays the value and status in the HMI

Cement Process

The family **Cement Process** is used to control the ratio of different fuels, interface of the gas analyser, and girth gear lubrication operation.

The following table describes the functions equipped according to the library:

Signal Type	Single Functions
MultiFuelCtrl	• controls the ratios of different fuels of different types (gas or liquid or solid) • calculates the percentage set point of each parameter continuously • monitors the process value feedback of individual fuel • controls the fuels output according to the operation modes (Auo 1, Auto 2, Manual and External) • generates alarms and alerts • displays the value and status in the HMI
GasAnalyser	• supervises and controls the gas analyser operation • configures the gas parameters for reading • defines the gas parameters which the analyser measures • displays the process value as output pin in the popup • generates alarms and alerts • displays the value and status in the HMI
GirthGearLubCtrl	• supervises and controls the girth gear lubrication operation • supports only Auto mode operation • generates alerts and alarms • displays the value and status in the HMI

Devices

The family **Devices** is used to manage Altivar variable speed drives ATV31M, ATV61M and ATV71M, TeSys T family of devices, Tesys T controller on CANopen, Modbus and Ethernet communication and energy meters in a communication based manner, and so on.

The following table describes the functions equipped according to the library:

Signal Type	Single Functions
CATV31M/61M/71M	• manages Altivar variable speed drives ATV31M/61M/71M in a CANopen communication based manner • controls the direction of rotation (forward/reverse) • enables the speed set point of the device • monitors the errors occured in the device • enables device reset by monitoring unit HMI • enables the user to use either the controller or the wired speed driver inputs or outputs to control either the direction of rotation or the speed set point • enables the owner (user) to select the control system operator or program that is responsible for setting the control
CTesyT	• manages the TeSys T family of devices in a communication based manner • controls the device operation • controls the forward/reverse operation of application • enables the change of speed (fast/slow) if you are working with a 2-speed motor start (double winding) • monitors the detection of errors that are associated with a device • enables to control or monitor the device
EATV31/61/71	• manages Altivar variable speed drives ATV31M/61M/71M in an Ethernet communication based manner • enables the speed set point of the device • selects the direction of the rotation (forward/reverse) • monitors the errors occurred in the device • enables the owner (user) to select the control system operator or program) that is responsible for setting the control • enables device reset by monitoring unit HMI
MBATV31/61/71	• manages Altivar variable speed drives ATV31M/61M/71M in a Modbus communication based manner • selects the direction of rotation (forward and reverse) • enables the speed set point of the device • monitors the errors occured in the device • enables the device resets by monitoring unit [HMI] • enables the user to use either the controller or the wired speed driver inputs or outputs to control either the direction of rotation or the speed set point • enables the owner (user) to select the control system operator or program that is responsible for setting the control
MBEM6400	• manages the energy meter MBEM6400 in a Modbus communication based manner • enables to reset the device • allows the most important device parameters to be monitored • displays the value and status in the HMI

Signal Type	Single Functions
MBEM7300	• manages the energy meter ION7300 in a Modbus communication based manner • enables to reset the device • allows the most important device parameters to be monitored • displays the value and status in the HMI
MBEM800	• manages the energy meter PM710/PM800 in a Modbus communication based manner • enables to reset the device • allows the most important device parameters to be monitored • displays the value and status in the HMI
MBTeSysT	• manages the TeSys T controller on Modbus communication and remains responsible for setting the control • controls the device operation • controls the forward/reverse operation of application • enables the change of speed (fast/slow), if you are working with a 2-speed motor start (double winding) • enables the resetting of the device by control and monitor unit • enables to control or monitor the device • displays the value and status in the HMI

For more information, refer to Device and Communication Manual.

Communication

The family **Communication** is used to provide information on Modbus technology (Modbus Messaging), CANopen technology (CANopen messaging) and Ethernet technology (Ethernet Messaging).

The following table describes the functions equipped according to the library:

Signal Type	Single Functions
ASRTPBdg	• Initiates requests to a device that is connected to an Advantys STB input/output island • enables read/write requests to be serialized to devices on the Advantys island with CAN clients • defines the timeout value that will be used for request end control • holds the vector for the RTP response zone • enables access to the I/O parameters and to the Advantys header status registers while the island is running • displays the value and status in the HMI
CClient	• enables select whether the parameter has read or write access • monitors transaction errors, identifying them on three levels in order to determine the source of the error • enables to define priorities for systems with multiple client accesses • obtains the transactions OK/NOK status, as well as their access times • displays the value and status in the HMI

Signal Type	Single Functions
CPortM	• enables the port in serializing and managing requests sent to a physical medium working on serial Modbus • defines the timeout value to be used in the DFB to activate the inoperable output variable • serializes CANopen requests to the controller physical port • displays the value and status in the HMI
CPortP	• enables the port in serializing and managing requests sent to a physical medium working on CANopen • enables the input rests output variable • defines the timeout value to be used in the DFB to activate the inoperable output variable • defines the slot in which the CPU is located on the rack • displays the value and status in the HMI
CScanner	• enables the internal client to generate several sequential requests to different memory positions in the remote device • displays the value and status in the HMI
EClient	• enables device data to be written or read to/from Ethernet technologies • enables select whether the parameter has read or write access • enables several consecutive registers to be read (available only with ModBusEthernetClient and ModBusClientBasic) • monitors transaction errors, identifying them on three levels in order to determine the source of the error • enables to define priorities for systems with multiple client accesses • obtains the transactions OK/NOK status, as well as their access times • displays the value and status in the HMI
EGtwMB	• functions as a bridge between serial devices and Ethernet buses • enables read/write requests to be issued to devices on a 485 bus through an Ethernet port (available only for ModBusClientBasic and ModbusScanner) • obtains the transactions OK/NOK status, as well as their access times • displays the value and status in the HMI
EPortM	• enables the port in serializing and managing requests sent to a physical medium working on serial Modbus TCP/IP with M340 PLC • displays the value and status in the HMI
EPortP	• enables the port in serializing and managing requests sent to a physical medium working on serial Modbus TCP/IP with Premium PLC • displays the value and status in the HMI
EPortQ	• enables the port in serializing and managing requests sent to a physical medium working on serial Modbus TCP/IP with Quantum PLC • displays the value and status in the HMI

Signal Type	Single Functions
EScanner	• refreshes read or write requests periodically for n registers issued to a single node on an Ethernet communications network • displays the value and status in the HMI
MBClientBs	• enables device data to be written or read to/from ModBus technologies • displays the value and status in the HMI
MBPortM	• enables the port in serializing and managing requests sent to a physical medium working on serial Modbus with M340 PLC • displays the value and status in the HMI
MBPortP	• enables the port in serializing and managing requests sent to a physical medium working on serial Modbus with Premium PLC • displays the value and status in the HMI
MBScanner	• uses an internal client to generate several sequential requests to different memory positions in the remote device • displays the value and status in the HMI
Statistics	• enables to assess the quality and speed of communications on a Modbus, Ethernet, or CANopen network • enables calculations to be performed on the client, scanner, port, and gateway blocks from the entire communications library and on devices in Modbus communications • allows the described totalizers to be reset • calculates the number of requests issued • communications information is calculated (maximum and minimum values, controller cycles, and so on) • communication times are calculated (request time, time of last request, and so on) • displays the value and status in the HMI

For more information, refer to Device and Communication Manual.

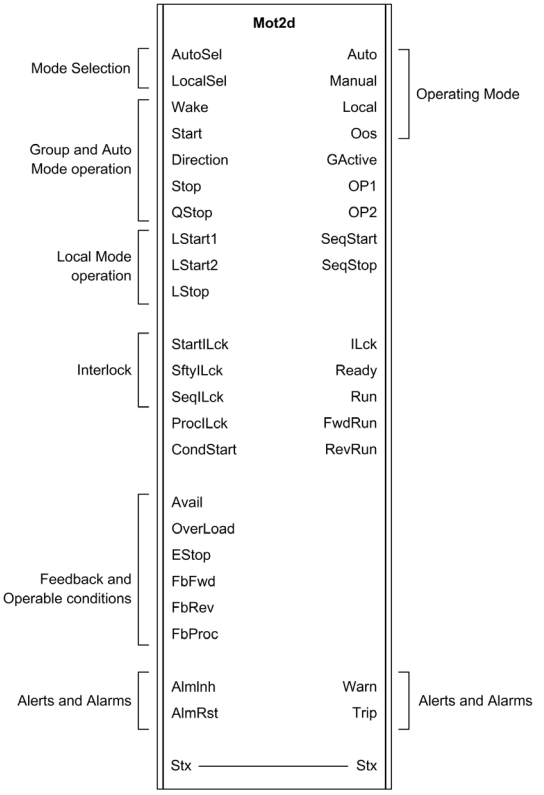
Function Blocks

3

Function Blocks

Function Block Interface

The function blocks have a common look and feel that facilitates easy usage and handling. The order of the pins is identical for most of the blocks as shown in the following figure.



Mode Selection

There are 1 maintenance and 3 operating modes.

This section allows to control mode of operations (Auto, Local, Manual and Oos). Auto and Local modes can be selected either from input pins of Unity Pro or from HMI control word. Manual and Oos modes can be controlled only from HMI control word.

Mode of Operation

The devices can be controlled by following 4 modes of operation:

Mode	Description
Auto	Auto mode operation of device can be controlled by `Start/Stop/QStop/Direction` input pins of Unity Pro. These input pins are connected either from `GrpCtrl` /`StrtMgt`/`StpMgt` /`RouteMgt`/previous device in the group.
Local	Local mode operation of device can be controlled by `LStart1`/`LStart2`/`LStop` input pin of Unity Pro. These input pins are connected from local IO panel **Start**/**Stop push** buttons.
Manual	Manual mode operation of device can be controlled from HMI by using control word. The single bit command from HMI, for example, `Stx.Ctw.xx` is passed on for operation of device.
Out of Service	This mode disables the operation of the block in all modes. The Out of Service mode is only available for actuator blocks, like motors or valves. It can be enabled from the **OOS** button available on the HMI (additional pop up of the block).

Interlock Types

There are 4 types of interlocks:

Interlock name	Parameter	Description
Starting Interlock	`StartILck`	This interlock is applicable only in Auto and Manual modes of operation. This interlock is considered only before starting of the device.
Safety Interlock	`SftyILck`	This interlock is applicable in all modes of operation. This interlock is considered in all conditions.
Sequential Interlock	`SeqILck`	This interlock is applicable only in Auto mode of operation. This interlock is considered in all conditions.
Process Interlock	`ProcILck`	This interlock is applicable only in Auto and Manual modes of operation. This interlock is considered in all conditions.

If the interlocks are active, then motor or actuators are available for operation.

Feedback and Operable Conditions

This block consists of feedbacks and operable conditions. For example, if motor is running and operable conditions (`Avail`, `Overload`, and `Estop`) are inoperable, motor is tripped. Similarly, if motor is started and feedbacks (`FbFwd`, `FbRev`, and `FbProc`) are not obtained, motor is tripped, and an alarm is generated.

Alarm Inhibition

This function allows to inhibit the alarms generated by the block based on the conditions configured to the input pin `AlmInh`. This can be used to prevent population of obvious alarms. For example, power trip in a plant can be conditioned to inhibit the alarms generated by the devices which gets tripped.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

The `AlmInh` parameter to inhibit alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

Alarm Reset

This function allows to reset the alarms either from input pin or from HMI.

Warning and Trip

This function gives user status of the block whether device has any warning or trip condition. For example, if motor is operational and any interlocks or feedback are inoperable, trip alarm is generated.

HMI Interface

All HMI data are transferred from the PLC to the HMI system in a most compact way. This means that a single bit of information is put on a 16 bit word each for status, alarm and control. The analog values of the process like (limit, delay, set point, and so on) are bundled in a structure and transferred to the HMI system.

Due to this concept only one pins of the block is necessary to interact with the HMI system. For example, `Stx.xx`, PLC - HMI, input/output pin which consist of analog values, status, alarm and control word.

All element of the structure except status word and alarm word can be written from the HMI. Communication can be done by the OPC server or the Modbus/TCP driver for Vijeo Citect.

Representation in the HMI

4

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Cement Library Template	42
HMI Object List	44
Popup Representation	47
Commonly Used Tabs	51

Cement Library Template

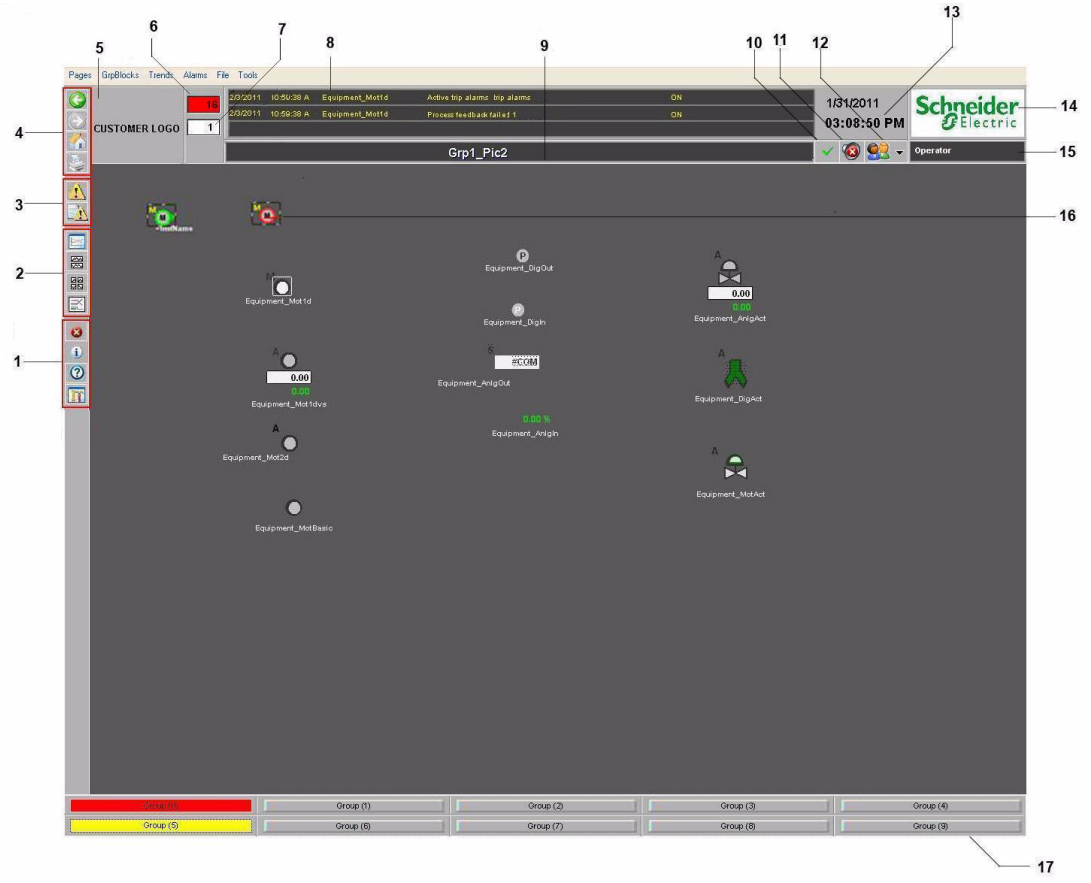
Description

The Cement Library template consists of objects to display the status of a detected alarm and an alert. It also displays the graphic details for Vijeo Citect HMI genie and popup pages. You can view the details about the operators if they are modifying any date or entering any command.

If any picture group is used, it is displayed in the HMI template. The status of unused pictures are grayed out.

Representation

The following figure shows the Cement Library template.



The following table shows the legend information of the Cement Library template:

Legend Names			
1 User tools	**6** Major alarm count	**11** Klaxon	**16** Genie
2 Trends	**7** Minor alarm count	**12** Login	**17** Group pages
3 Alarms	**8** Top 3 alarms	**13** Date and time	-
4 System tools	**9** Page name	**14** Schneider Electric logo	-
5 Customer logo	**10** Acknowledge all alarms	**15** User name	-

The display of gray status for unused picture groups is user defined. You can use upto 10 groups. Each group can have the pages as configured by the user within it.

HMI Object List

HMI Representation

Each function or object in the HMI is represented by a status display (Vijeo Citect: Genie), indicating the status and appropriate values. By clicking the **Status display**, a popup opens with additional information for operator control.

Several tabs are available depending on the single functions. The HMI interface of the function block provides the necessary animation for status display and popup.

Navigation Toolbar

Toolbar Buttons	Symbol	Description
Back		Navigates to the back page.
Front		Navigates to the next page.
Home		Navigates to the home page.

Button Labelled Alarms

Alarm Buttons	Symbol	Description
Active alarms		Navigates to the active alarm page.
Alarms summary		Navigates to the alarms summary page.

Button Labelled Trends

Trend Buttons	Symbol	Description
Single trend		Opens the single trend popup.
Double trend		Opens the double trend popup.

Trend Buttons	Symbol	Description
Configure trend groups		Opens the popup to configure trend groups
Process analyst		Opens the process analyst.

Button Labelled Tools

Tool Buttons	Symbol	Description
Printer		Prints the current page.
Close		Closes the Vijeo Citect window.
Hide/Unhide		Hides/unhides the instance name in the page.
Help		Opens the online help of Vijeo Citect.
Tools		Opens the admin tools page, and provide the options to tag debug in the Vijeo Citect. This page displays the folllowing options: • Citect kernel • Citect configurations • System/hardware • Citect information • IO server • Time keeping

Graphical Objects

Serial Number	Graphical Objects	Different Sets			Development in version
		Small	Medium	Large	
1	Bucket Elevator	YES	YES	YES	1.0
2	Dust Collector	YES	NO	NO	
3	Bag House	YES	YES	YES	
4	Belt Conveyor	YES	YES	YES	
5	Air Slide (Pipe Conveyor)	YES	YES	YES	
6	Ball Mill (Cement Mill)	YES	YES	YES	
7	ESP	NO	YES	YES	
8	VRM (Raw Mill/Coal Mill)	NO	YES	YES	
9	Cyclone	NO	YES	YES	
10	PreCalciner	NO	YES	YES	
11	Kiln	YES	YES	YES	
12	Fire Flame object for flame sensor and detection - on/off status	YES	NO	NO	
13	Damper	YES	NO	NO	
14	Shut Off - Gate	YES	YES	NO	
15	Stacker	NO	YES	YES	
16	Weigh Feeder	NO	YES	YES	
17	Roller Press	NO	YES	YES	
18	Rotary Air Lock	NO	YES	NO	
19	Diverter	YES	NO	YES	
20	Emission - Gas Objects (Chimney)	YES	YES	NO	
21	Silo (for Homogenization/Blending Silo and Cement Silo)	YES	YES	YES	
22	Screw conveyor	YES	YES	YES	
23	Water Sprinkle object for Water Spray System	YES	YES	YES	

Popup Representation

Tab Selection Buttons

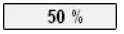
Item	Symbol	Description
Operator tab		Displays the **Operator** tab of the control module.
Extended tab		Displays the **Extended** tab of the control module.
Configuration tab		Displays the **Configuration** tab of the control module.
Interlock tab		Displays the **Interlock** tab of the control module.
Alarm tab		Displays the **Alarm** tab of the control module.
Trend tab		Displays the **Trend** tab of the control module.
Information tab		Displays the **Information** tab of the control module.
Analog tab		Displays the analog values.
Digital tab		Displays the digital conditions.
Starting interlock sub-tab		Displays the starting interlock status and description of the control module.
Safety interlock sub-tab		Displays the safety interlock status and description of the control module.
Sequential interlock sub-tab		Displays the sequential interlock status and description of the control module.
Process interlock sub-tab		Displays the process interlock status and description of the control module.
Schedule interlock sub-tab		Displays the schedule interlock status and description of the control module.

Item	Symbol	Description
Analog 1 sub-tab		Displays the analog values of the control module.
Analog 2 sub-tab		Displays the analog values of the control module.
Digital 1 sub-tab		Displays the digital conditions of the control module.
Digital 2 sub-tab		Displays the digital conditions of the control module.
Digital 3 sub-tab		Displays the digital conditions of the control module.

NOTE: Not available buttons are grayed (dimmed).

Buttons, Lines and Fields

Description	Symbol	Meaning
Auto mode		Selects the Auto mode of the control module.
Manual mode		Selects the Manual mode of the control module.
Local mode		Selects the Local mode of the control module.
Out of Service mode		Selects the Out of Service mode of the control module.
External mode		Selects the External mode of the ratio controller control module.
Cascade mode		Selects the Cascade mode of the PID control module.
Alarm mask		Masks the trip alarms of the control module.
Warning mask		Masks the alert of the control module.

Description	Symbol	Meaning
Trend popup		Opens the associated **Trend** window of the control module.
Alarm page navigation		Opens the associated **Alarm** window which contains the filtered alarms and alerts.
Alarm acknowledge		Acknowledges the alarms of the control module.
Alarm reset		Resets the alarms of the control module.
Note acknowledge		Acknowledges the note written by the operator for the control module.
Group available/Route selection		Gives command to start group/select route in the control module.
Start/ON		• Starts the motor • **On** command of the control module
Stop/OFF		• Stops the motor or the group • **Off** command of the control module
Quick stop		Gives a quick stop command (**QStop**) in the control module.
Bypass/Hold		• Bypasses the process value • Holds the sequence of the control module
Ramp down to Low limit		Ramps down the value to low limit in the control module.
Ramp down by step 1		Ramps down the value by step 1 in the control module.
Ramp up by step 1		Ramps up the value by step 1 in the control module.
Ramp up to high limit		Ramps up the value to high limit in the control module.
Value field (gray background)		Displays the value which is not editable.
Value field (white background)		Displays the value which is editable.

Description	Symbol	Meaning
General rule		In all popups, buttons are displayed in gray color.
Major alarm count		Displays the number of major alarms.
Minor alarm count		Displays the number of minor alarms/trips.
Page name		Displays the page name/function block name.
Klaxon		Displays the voice alarm (including alarming and silencing status).
Login		Displays the login symbol for the user.
Genie		Displays the dynamic objects which are available in the page.
Group page		Displays the group pages, which the user can configure.

Commonly Used Tabs

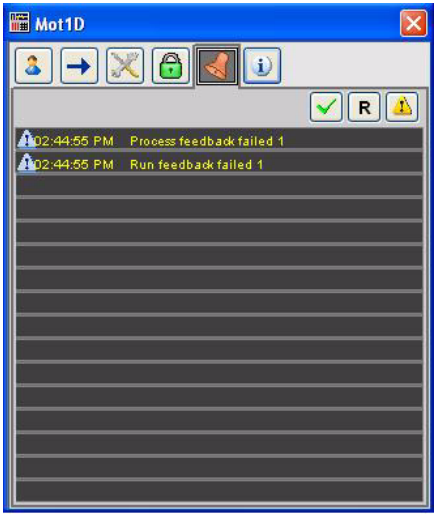
Overview

This provides information about the commonly used tabs and sub-tabs. These tabs are commonly available as popups in most of the function blocks.

The commonly used tabs are:

- **Alarm** tab
- **Information** tab
- **Interlock** sub-tabs

Alarm Tab



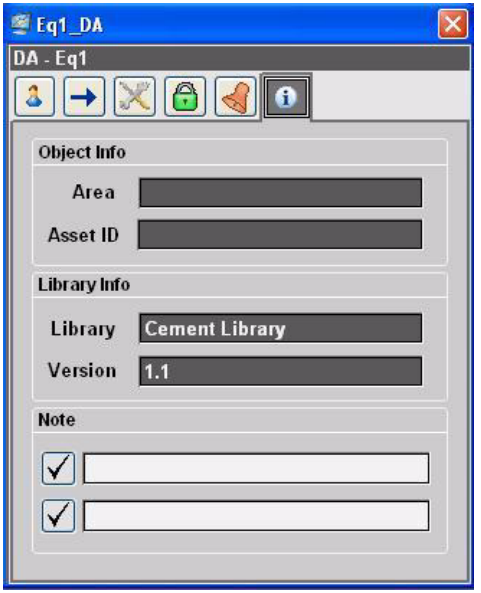
In this tab, all the alert and trip alarms of this block will be displayed with date, time and tag name.

For the description about available tabs and sub-tabs, please refer to Tab Selection Buttons *(see page 47)*.

The following table describes the **Alarm** tab:

Symbol	Description
	All alarms of this block are acknowledged.
	All alarm conditions are reset.
	Navigates to the alarm summary page.

Information Tab



This tab has the following sub-sections:

- **Object Info**
- **Library Info**
- **Note**

The following table describes the sub-section **Object Info**:

Item	Description
Area	Displays the area to which the object belongs. This is defined during the instantiation of the block and cannot be edited in the popup.
Asset ID	Displays the asset ID of the object. This is defined during the instantiation of the block and cannot be edited in the popup.

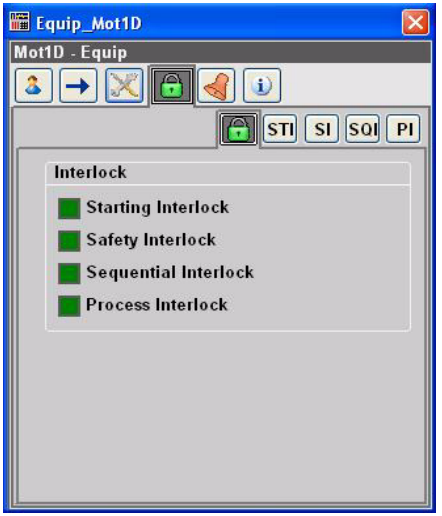
The following table describes the sub-section **Library Info**:

Item	Description
Library	Displays the library name. This is a static text.
Version	Displays the library version. This is a static text.

The following table describes the sub-section **Note**:

Symbol	Description
	The operator can write 2 short notes. It can be acknowledged by clicking the button available before the text. This acknowledgement erases the note.

Interlock Sub-tabs



The following sub-tabs are commonly used in all the function blocks:

- Interlock
- **STI**
- **SI**
- **SQI**
- **PI**

The following table describes the **Interlock** sub-tabs:

Symbol	Description
	Navigates to main interlock page.

Item	Description
STI	All the starting interlock conditions are displayed with description. Also, individual interlock condition can be bypassed in this section.
SI	All the safety interlock conditions are displayed with description. Also, individual interlock condition can be bypassed in this section. **NOTE:** The figure for **Safety Interlock** is same as that for Starting Interlock.
SQI	All the sequential interlock conditions are displayed with description. Also, individual interlock condition can be bypassed in this section. **NOTE:** The figure for **Sequential Interlock** is same as that for Starting Interlock.
PI	All the process interlock conditions are displayed with description. Also, individual interlock condition can be bypassed in this section. **NOTE:** The figure for **Process Interlock** is same as that for Starting Interlock.

Using the Cement Library

5

Overview

This chapter describes the features of the Cement Library.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Operator Logging	56
Alarm Logging and Alarm Summary	57
Dim/Unavailable Buttons	58
Alarms in the Popup	59
Access Level Support	60
Enable Use of Double Monitor	62
Acknowledge Alarms	65
Local Language Support	66
Navigation to Trend and Alarm Window from the Popup	68
Trend Assignment	69
Color Customization	70
Navigation Buttons	73
Information Popup	74
Out of Service Mode	75

Operator Logging

Introduction

Operator logging file is available in the *Data* folder of the Vijeo Citect installation path with the file name *Cem_OperLog.DBF*. To view this file, open it using Microsoft Office Excel tool.

The data will be displayed in excel form and contains the following information:

Item	Description
DATE	Date value on value change
TIME	Time value on value change
USERNAME	Logged in user who has changed the value
LOCATION	Location where the value has changed. It is displayed as Full Control Module Name with the format that has been set in UAG
OPACTION	Log message about the action
OLD VALUE	Value before it is changed
NEW VALUE	Value after it is changed

Old value and new value will be present only when there is a change in the analog parameters.

For the command operations, message is logged as Auto Mode Initiated, Manual Mode Initiated, and so on. Those buttons having multiple functions (like OOS), message is logged as OOS Active (whenever OOS is enabled) and OOS Inactive (whenever OOS is disabled) in the **OPACTION** field.

The following figure shows the operator log file.

	A	B	C	D	E	F	G
1	DATE	TIME	USERNAME	LOCATION	OP ACTION	OLDVALUE	NEWVALUE
2	2/2/2009	10:50:01 AM	a	eq2_mul	MV Changed	10	25
3	2/2/2009	10:50:06 AM	a	Eq4_Add4	MV Changed	12	24
4	2/2/2009	10:50:09 AM	a	Eq4_Add4	MV Changed	24	36
5	2/2/2009	01:40:49 PM	a	eq2_mul	MV Changed	25	55
6	2/2/2009	02:25:26 PM	a	Eq3_Anain	MV Changed	0	24
7	2/2/2009	02:38:22 PM	a	Eq3_Anain	MV Changed	24	77
8	2/2/2009	02:38:32 PM	a	eq2_mul	MV Changed	55	9
9	2/2/2009	02:38:54 PM	a	eq2_mul	MV Changed	9	21
10	2/2/2009	02:47:06 PM	a	Eq4_Add4	MV Changed	36	66
11	2/2/2009	03:42:54 PM	a	Eq4_Add4	MV Changed	66	88
12	2/2/2009	03:47:25 PM	a	Eq3_Anain	MV Changed	77	29

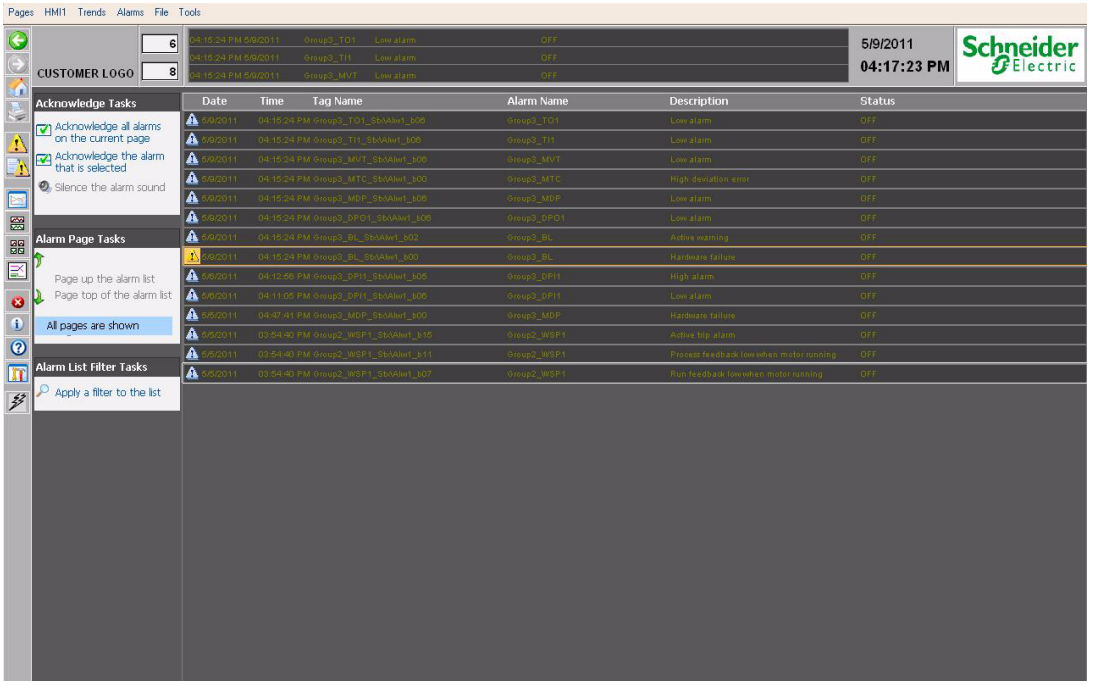
Alarm Logging and Alarm Summary

Introduction

Alarm log file and alarm summary log files are available in the *Data* folder of the Vijeo Citect installation path with the file names *Alarmlog* and *AlarmSummarylog*. To view these files, open it using Microsoft Office Excel tool.

The data will be displayed in excel form and contains the following information:

Item	Description
DATE	Date value on alarm occurence
TIME	Time value on alarm occurence
NAME	Name of the alarm
TAGNAME	Alarm tagname of the instance
DESCRIPTION	Description of the alarm occured
LOGSTATE	State of the alarm occured

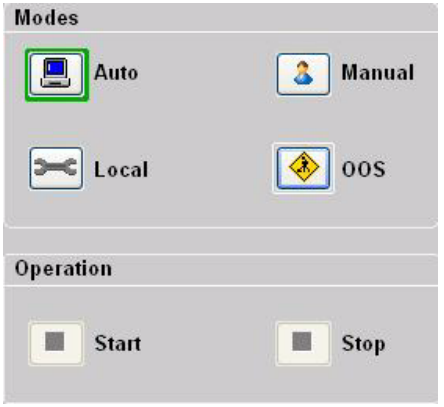


Dim/Unavailable Buttons

Introduction

If the user login has got insufficient privilege, the button and text entry field is locked for user operation. The locked button appears dimmed because it is unavailable for user operation.

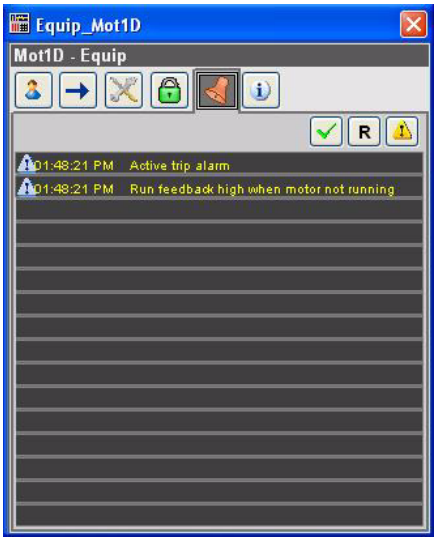
For example, in the following figure, the **Start** and **Stop** buttons are dimmed (unavailable).



Alarms in the Popup

Introduction

The alarms of the instantiated object are available in the **Alarm** tab of its popup.



The alarm display in the popup have the following details:

Item	Description
TIME	Time value on alarm occurrence
TAG DESCRIPTION	Alarm tag name of the instance

An alarm tag name upto 32 characters is supported. A name beyond 32 characters is truncated.

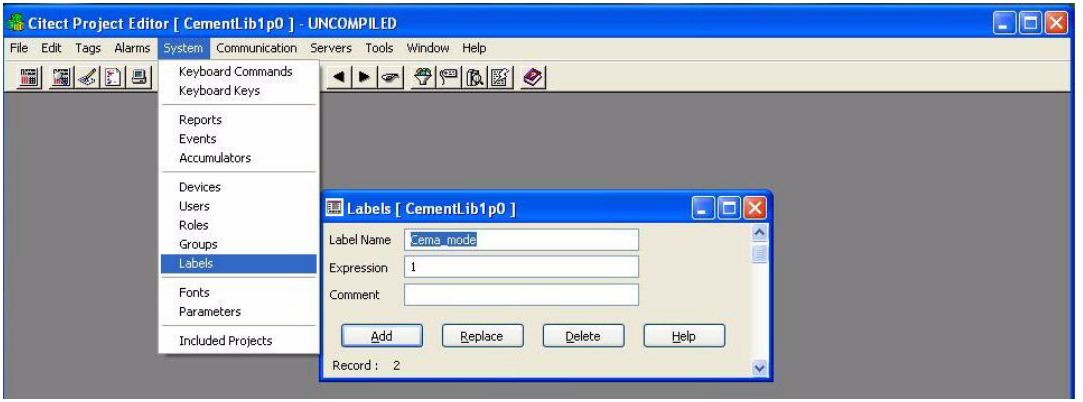
Access Level Support

Introduction

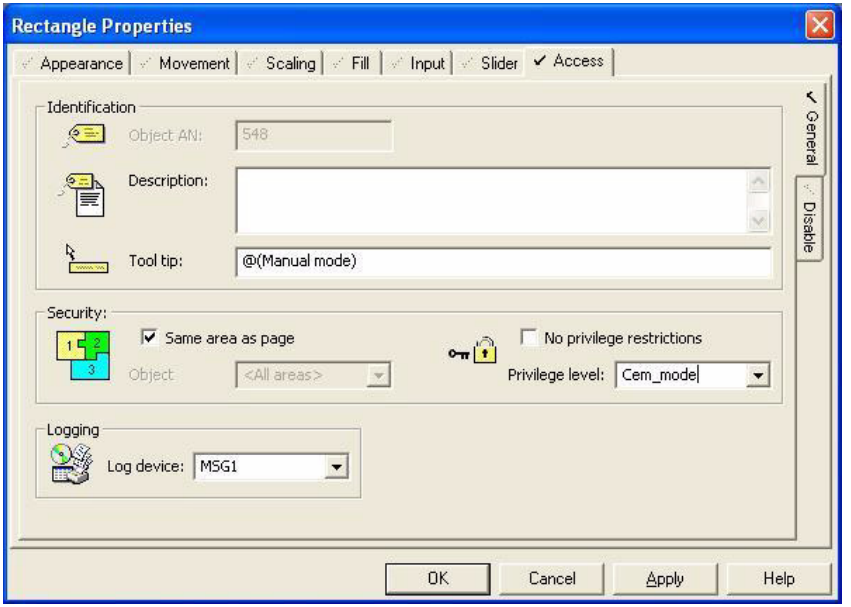
Privilege for the login users is defined in terms of labels. Several standard labels with different predefined access levels are defined in the library as shown in the following figures.

Whenever a privilege for a group of objects need to be changed, modify the **Expression** field in the appropriate label defined for the privilege in the library.

The name of the label is defined in the **Label Name** field. The **Expression** field, present in the label indicates the privilege level. For example, if a dynamic object like button is assigned with the label **Cem_mode** in **Privilge level** field, that object will have privilege of level 1.



These labels are then assigned at the **Privilege level** field in the **Access** tab for all those dynamic objects which have privileges assigned like button and so on, as shown in the following figure.



To modify the privilege, you need to replace the existing privilege number with the new privilege number in the **Expression** field of the label, before Vijeo Citect goes to Run mode.

The following is the list of labels that are made available in the library. These labels are then used in each library blocks in HMI.

Serial Number	Label name	Privilege Level	Description
1	Cema_mode	1	To set the operation mode like Auto or Manual
2	Cema_operate	1	To operate a device like Start, Stop, and so on
3	Cema_value	1	To set operating set point or values
4	Cema_reset	1	To reset faults and alarms
5	Cema_limits	3	To modify alarm limits like HH, H, L, and LL
6	Cema_masking	3	To mask alarms and alerts
7	Cema_service	3	To execute service like OOS and so on
8	Cema_configure	5	To configure devices like PIDs

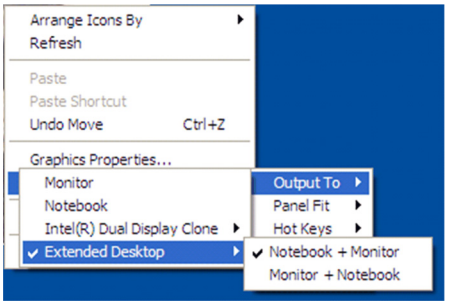
Enable Use of Double Monitor

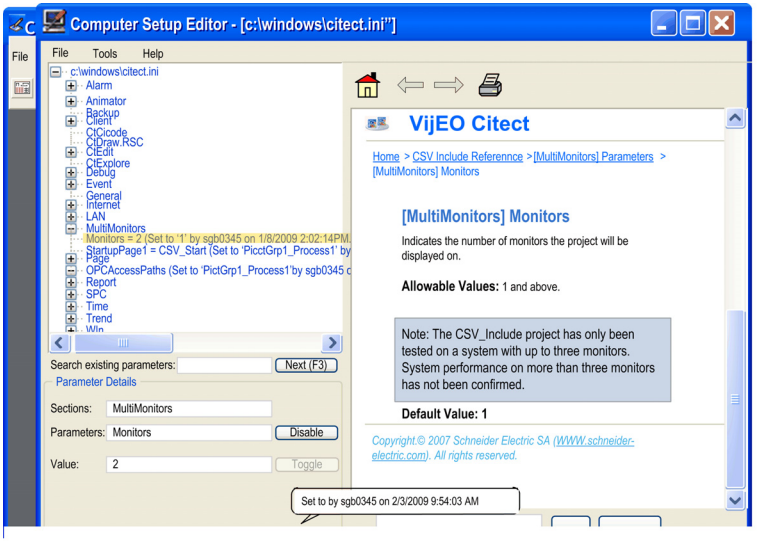
Introduction

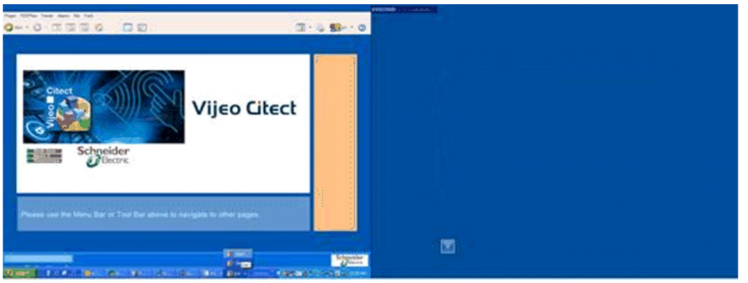
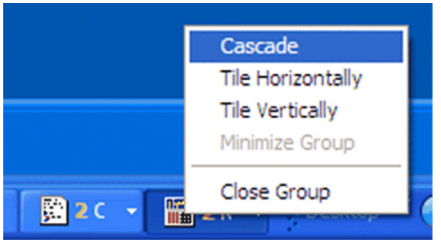
Multi monitors can be implemented by

- using the extended monitor property available on inserting a graphic card in the system (Desktop PC), or
- adding an additional monitor with the notebook monitor.

Default cicodes are defined in the library which make it possible to use the process library for the double monitor feature.

Step	Action
1	Insert a graphics card in the desktop PC or connect the additional monitor to the notebook. Right-click the desktop and select the **Graphic output** →**Output To** → **Extended Desktop** display. 

Step	Action
2	In Vijeo Citect **Computer Setup Editor**, set the multiple monitor parameters, defining the number of monitors to be used and the resolution of each monitor. A maximum of 6 monitors can be configured. 

Step	Action
3	When you run the Vijeo Citect application, 2 runtime start windows are shown. Right-click the **Start Windows** and select the **Cascade** to display both the runtime windows.   **NOTE:** You can also define different start pictures for different monitors.

NOTE: Different pages and different popups can be opened on both the primary or extended monitor.

Acknowledge Alarms

Introduction

To acknowledge all alarms:

- In the template, click **Acknowledge all alarms** button.
- In the **Alarm** tab of the popup, click the **Acknowledge** button .

This will acknowledge all alarms and alerts. In addition, the detected errors are reset, clearing the inactive alarms from the list.

Local Language Support

Introduction

All HMI elements of the blocks are prepared so that they support language localization. For successful localization, language based DBF file should be present. Standard localization languages like English, French, Spanish, and German languages are supported.

The language DBF files are located in the project. Here you need to enter the translated meaning for the information present in the **NATIVE** column into the **LOCAL** column present inside the DBF, before Vijeo Citect is put into Run mode.

The different language files that you need to use are as follows:

You need to make an entry in the **LOCAL** column with the reference from **NATIVE** column.

	A	B
1	NATIVE	LOCAL
2	Enter Old Password	
3	Alarm Summary	
4	Control Mode	
5	Write	
6	Integral	
7	Configure Trend Groups	
8	Trend Name	
9	Configure Alarm Groups	
10	Network page	
11	Display Citect.ini config file	
12	Free Disk Space	
13	Select	
14	Time will be set on all computers to this computer's time	
15	Invalid Category	
16	Trend Duration	
17	OnDateTime	
18	Display page name.	
19	Number of Write Requests	
20	Trend Selection	
21	Previous page	
22	Disabled Alarm	
23	No of Report Connections	
24	End Time	
25	Select trend tag	
26	Set Trend Span	
27	Citect run time	
28	Legend	
29	Full Name	
30	Active Alarm Filter	
31	!TrendStats	
32	Display All Alarms	
33	This page has been disabled	
34	Trend Date/Time	

Navigation to Trend and Alarm Window from the Popup

Introduction

When you click the **Alarm** button in the **Alarm** tab of the popup, the alarm page is opened.

When you click the **Historical Trend** button in the **Trend** tab of the popup, the trend popup is opened.

All trend tags of the instance are automatically passed to this trend popup.

Trend Assignment

Default Method in Vijeo Citect

To compare the trend tag of one instance/block with trend tag of another instance/block, open the trend popup from the template *(see page 42)*. You have to manually assign the trend pens.

Easy Trend Assignment Feature of the Library

The analog parameters have trend tags. For example, If `PV` value of `PID` and `APV` of `AnlgIn` have to be compared, right-click the `PV` **Process Value** display in `PID` block popup and `APV` **Process Value** display in `AnlgIn` block popup. The trend popup opens with the process value of `PID` and `AnlgIn` as trend pens.

Color Customization

Introduction

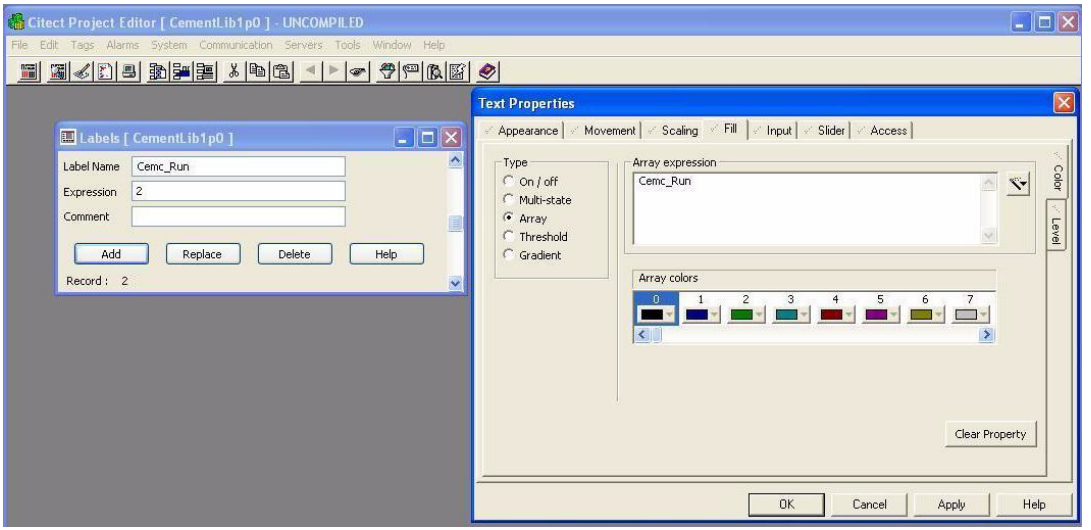
Color customization gives you the ability to define custom colors for the popups and the objects like group frames, outer frames, editable boxes, and so on present in the genie and popups.

You can apply the color customization functionality at the following levels:

Serial Number	Color Customizable Objects Label	Description
1	Cemc_AlOnUnack	Active trip indication in genie.
2	Cemc_WaOnUnAck	Active alert indication in genie.
3	Cemc_AlOnAck	Active and acknowledged trip indication in genie.
4	Cemc_WaOnAck	Active and acknowledged alert indication in genie.
5	Cemc_AlOffUnack	Inactive and unacknowledged trip indication in genie.
6	Cemc_WaOffUnAck	Inactive and unacknowledged alert indication in genie.
7	Cemc_AlWaMask	Trip/alert masked indication in genie.
8	Cemc_Sleep	Genie/object is in sleep mode.
9	Cemc_Awake	Genie/object is in awake mode.
10	Cemc_Run	Motor RUN indication in genie.
11	Cemc_Stop	Motor STOP indication in genie.
12	Cemc_Open	Actuator OPEN indication in genie.
13	Cemc_Close	Actuator CLOSE indication in genie.
14	Cemc_ON	Digital ON indication in genie.
15	Cemc_OFF	Digital OFF indication in genie.
16	Cemc_Mode	Mode text, that is, A, M, L, O, and S indication in genie.
17	Cemc_PV	Color of the process value text indicated in the `PID`, `AnlgIn`, `MBATV`, `EATV`, `CATV`, `Mot1dVS`, `Mot2dVS` genie and popup.
18	Cemc_SP	Color of the set point text indicated in the `PID`, `AnlgOut`, `MBATV`, `EATV`, `CATV`, `Mot1dVS`, `Mot2dVS` genie and popup.
19	Cemc_Editbox	Edit box where you can enter the data.
20	Cemc_NonEditBox	Non editable box where the data is displayed.

Serial Number	Color Customizable Objects Label	Description
21	Cemc_FeedBackSt	Feedback status indication in the popup, for example, run forward or run backward or pressure switch feedback and so on.
22	Cemc_InterlockSt	Interlock status indication in the popup, for example, starting interlock or safety interlock or sequential interlock or process interlock.
23	Cemc_BypassSt	Bypass status of group start/stop management indication in the popup.
24	Cemc_DigIn	Digital input indication in the genie.
25	Cemc_DigOut	Digital output indication in the genie.
26	Cemc_HHLL	The text color indication in genie of analog input block when HighHigh/LowLow value is reached.
27	Cemc_HL	The text color indication in genie of analog input block when High/Low value is reached.
28	Cemc_RatioEn	Ratio enabled status indication in the popup of ratio controller.
29	Cemc_RatioDis	Ratio disabled status indication in the popup of ratio controller.
30	Cemc_RatioDis	Ratio disabled status indication in the popup of ratio controller.
31	Cemc_RunWOFb	Genie/equipment is running without feedback.
32	Cemc_OP1noFb	Genie/actuator is running without feedback.
33	Cemc_HFailAck	Hardware inoperable acknowledged indication in genie.
34	Cemc_HFailUnAck	Hardware inoperable unacknowledged indication in genie.

There are 1 million colors available in Vijeo Citect HMI-SCADA. However, the color palette in Vijeo Citect is limited to 256 colors which is shown in the following figure.



NOTE: Standard labels defining the color code will be used.

Example

The label name **Cemc_Run** is defined in the properties where the color customization for **Run** command is required. For example, the label name for motor run color indication, **Cemc_Run** is given in the **Array expression**. The label **Cemc_Run** has an expression **2**. Hence, in normal conditions, the motor run color is green (The **Array colors** 2 = green in the above shown color palette). If there is a need to change this motor run color from green to blue, you have to change the label **Cemc_Run** expression from 2 to 3 (The **Array colors** 3 = blue in the above shown color palette).

Navigation Buttons

Introduction

Navigation buttons with the names of group defined in UAG will be displayed in Vijeo Citect at runtime. Upto 10 groups are supported in this library. When you click the button, pages under the corresponding group will be displayed. You can navigate to those pages by selecting the page from the list.

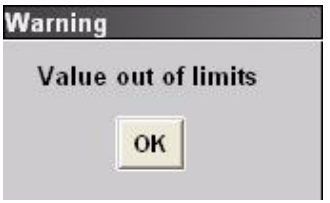
The following table describes the different animations for the navigation buttons:

Animation	Meaning
Flashing yellow	There is an active alert in the group.
Steady yellow	There is an active and acknowledged alert in the group.
Flashing red	There is an active trip in the group.
Steady red	There is an active and acknowledged trip in the group.

Information Popup

Introduction

When an operator enters a value beyond the limits defined (HIGH scale and LOW scale), the following information popup is displayed.



Out of Service Mode

Introduction

The Oos mode is used to avoid starting of the motor during maintenance activities. When the motor is in stopped state, the motor will be changed to Out of Service mode by the operator through the popup.

When the motor is in Oos mode, the motor cannot be started by the operator or through the group operation. The operator again changes the mode of the motor from Oos mode to Auto/Manual/Local.

NOTE: When the motor is in running state, then the Oos mode selection is grayed out.

Implementation Paradigms

6

Overview

This chapter describes the implementation paradigms common to all the function blocks.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Defining Startup Function in Vijeo Citect	78
Scaling of Values for Visualization	88
Eng. Min/Eng. Max Property of the Control Module	89

Defining Startup Function in Vijeo Citect

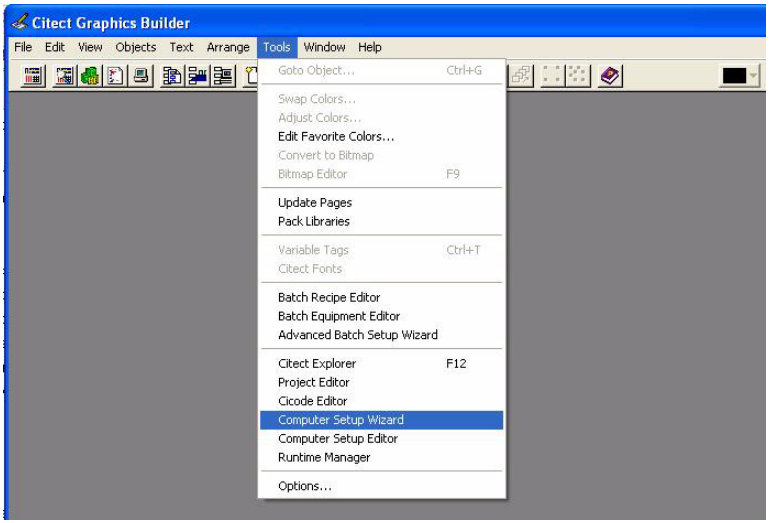
Startup Function in Vijeo Citect

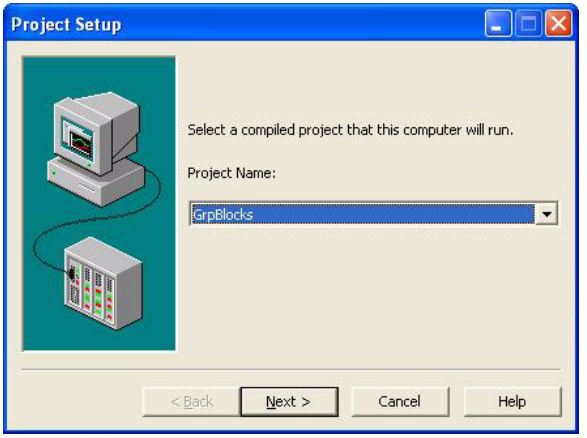
Defining the startup function in Vijeo Citect involves the following 3 basic steps:

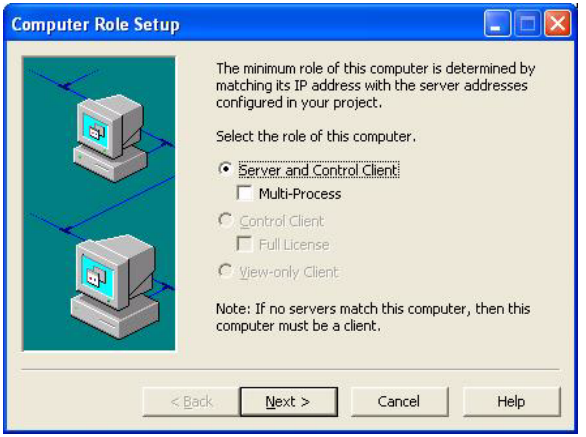
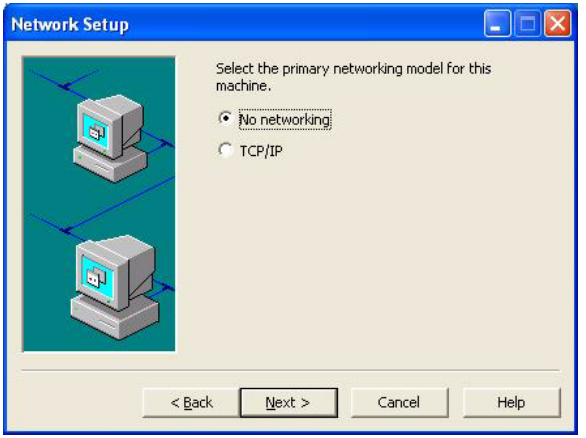
1. Selecting the appropriate configurations based on the project.
2. Modifying the selected startup function.
3. Configuring the parameters according to the project.

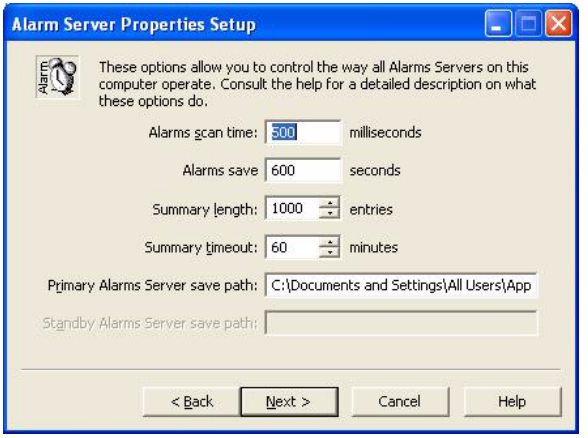
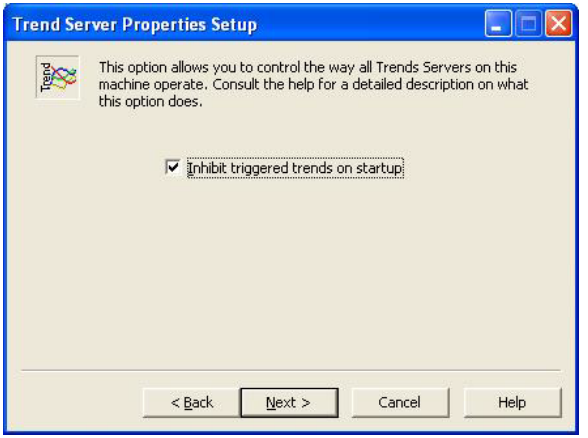
These basic steps are detailed in the following procedures:

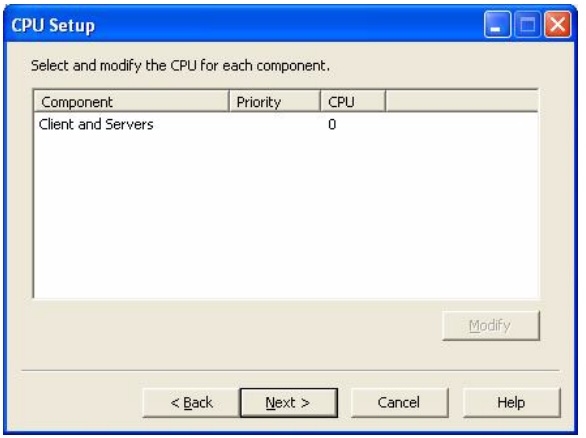
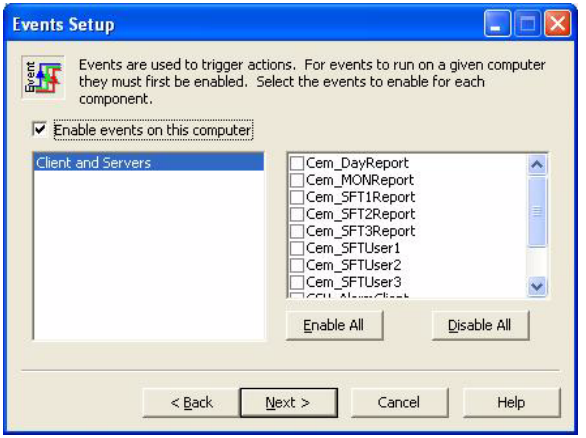
Step 1: Selecting the appropriate configuration based on the project.

Step	Action
1	From the **Citect Graphics Builder** screen, click **Tools** →**Computer Setup Wizard**.  **Result**: The **Citect Computer Setup Wizard** dialog box appears.

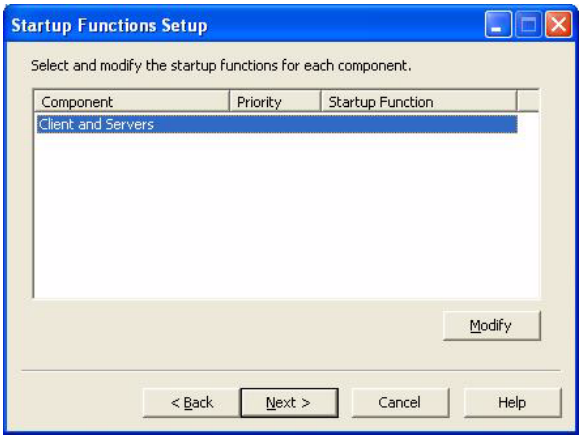
Step	Action
2	Select **Custom Setup** and click **Next**. **Result**: The **Project Setup** window appears.
3	Select the relevant project in the **Project Name** list and click **Next**.  **Result**: The **Computer Role Setup** window appears.

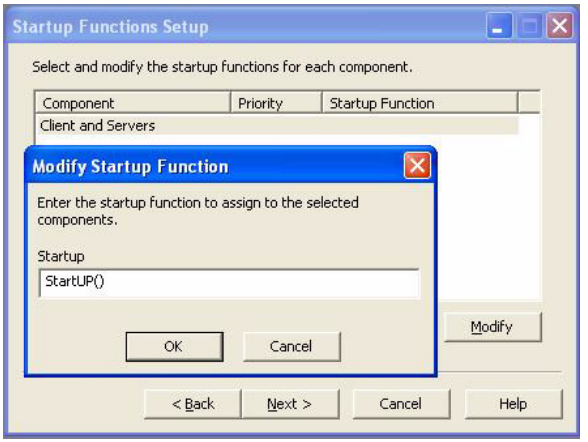
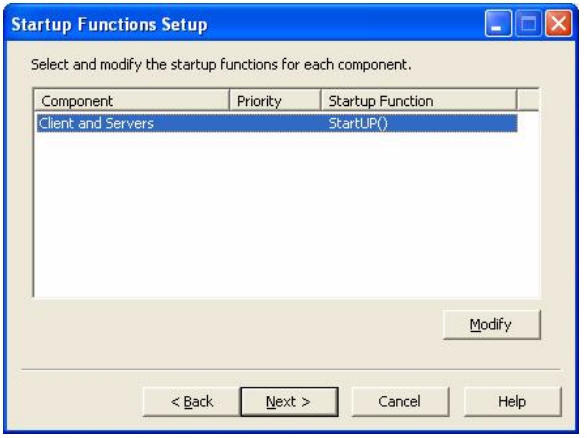
Step	Action
4	Select the appropriate configurations in the series of windows based on the project and click **Next**.  **Result**: The **Network Setup** window appears.
5	Select the primary networking model for the machine. Click **Next**.  **Result**: The **Alarm Server Properties Setup** window appears.

Step	Action
6	Enter the alarm related values to control the Alarm Server and specify the path to save the primary alarms. Click **Next**.  **Result**: The **Trend Server Properties Setup** window appears.
7	Check the **Inhibit triggered trends on startup** check box. Click **Next**.  **Result**: The **CPU Setup** window appears.

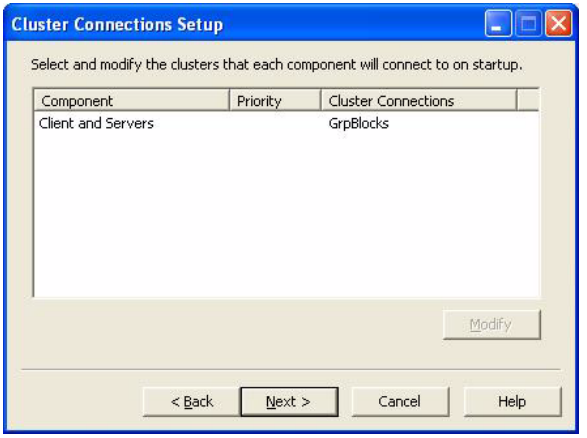
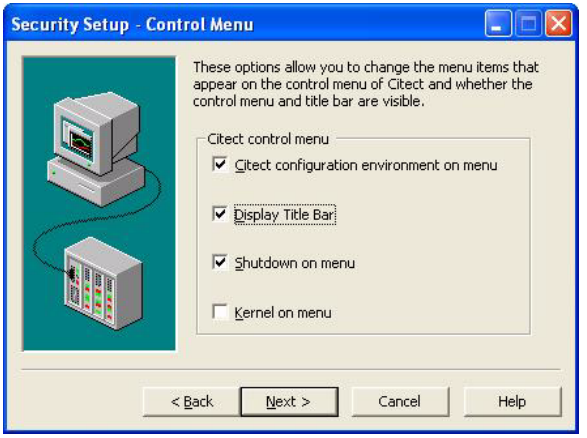
Step	Action
8	Select the component for modifying the CPU. Click **Next**.  **Result**: The **Events Setup** window appears.
9	Check the **Enable events on this computer** check box. To enable the events you need, select the check boxes in the list on the right hand side. Click **Next**.  **Result**: A **Startup Functions Setup** window (shown in step 2) appears that enables modification of startup functions.

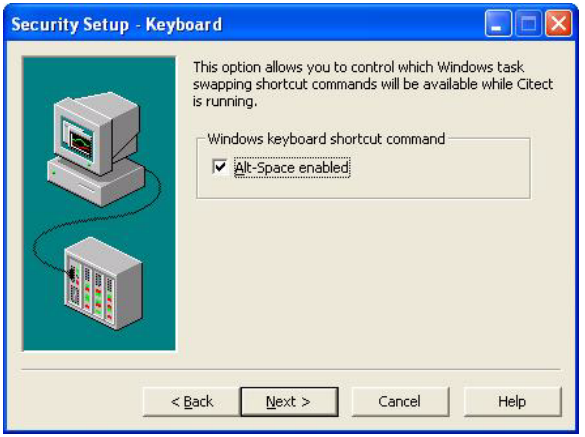
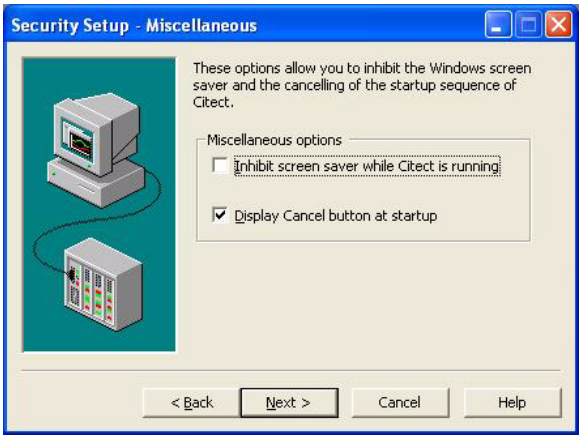
Step 2: Modifying the selected startup function.

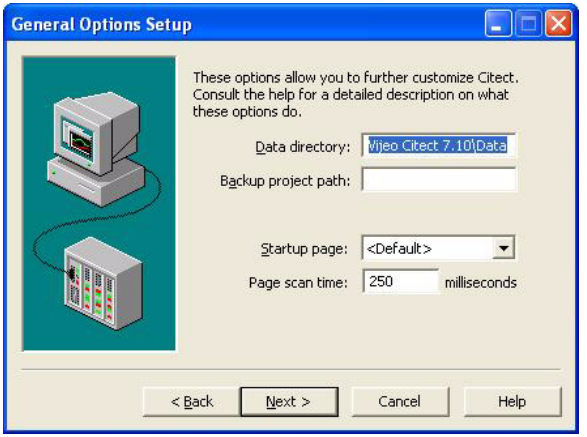
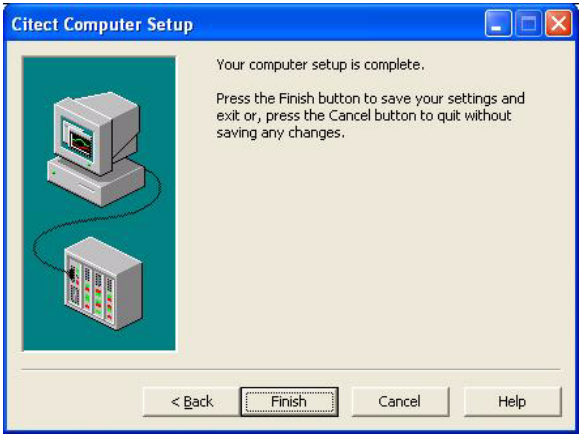
Step	Action
1	From the **Startup Functions Setup** window, select the component for modifying the startup function and click **Modify**.  **Result**: The **Modify Startup Function** popup window appears.

Step	Action
2	Define the startup function as `StartUp()`. Click **OK**.  **Result**: The **Modify Startup Function** popup window disappears, retaining the **Startup Functions Setup** window.
3	Click **Next**.  **Result**: A series of window (shown in the next step) appears to configure the parameters for computer setup. The first window is the **Cluster Connections Setup** window.

Step 3: Configuring parameters for computer setup. This setting is done only once during the project creation or until it is manually changed.

Step	Action
1	In the **Cluster Connections Setup** window, select and modify the clusters that each component connects to on startup. Click **Next**.  **Result**: The **Security Setup - Control Menu** window appears.
2	Check the required options in the **Citect control menu** area. Click **Next**.  **Result**: The **Security Setup - Keyboard** window appears.

Step	Action
3	Click **Alt-Space enabled** in the **Windows keyboard shortcut command** area. Click **Next**.  **Result** • The checked option allows the swapping shortcut commands while Vijeo Citect is running. • The **Security Setup - Miscellaneous** window appears.
4	Select the required check boxes to inhibit windows screen saver and to display **Cancel** button at startup. Click **Next**.  **Result**: The **General Options Setup** window appears.

Step	Action
5	Enter the general details for further customizing the Vijeo Citect. Click **Help** for a detailed description. Click **Next**.  **Result**: The **Citect Computer Setup** window appears to display that the computer setup is completed.
6	Click **Finish** to quit by saving the settings or click **Cancel** to quit without saving the changes.  **Result**: The computer setup parameters are configured.

Scaling of Values for Visualization

Overview

Scaling of values for visualization in the HMI is done in the DFB but not in the HMI. An already scaled value can be rescaled by the standard Vijeo Citect mechanism.

Scaling Values in the DFB

The variables of the type `IO_PLC` are scaled in Unity Pro (DFB) from raw count values (depending on the type of the IO module) to `Eng.values` by using scaling blocks.

For example, in the `AnlgIn` *(see page 161)* block, the output `PV` is calculated in the Auto mode from the input value by using scaling values.

The scaling values are based on the following `PAR.Type` values:

- The value `PAR.Type` defines the different conversion types and scaling values or the conversion to a `REAL` type only (16#FF).
- If conversion and scaling are activated, `PV` is calculated by using `PAR.Min` and `PAR.Max`.
- `PAR.RawMin` and `PAR.RawMax` are used for generic devices (16#01).

The input `V` is converted into a real value and given to the output `PV` for conversion into a `REAL` type (16#FF).

If the operation mode is changed from Auto to Manual mode, the value `HMI.MV` that is entered in the HMI automatically replaces the value `PV`.

When changing from Manual mode, the value `PV` is calculated again. Additionally, the input `RV` replaces the value `PV` if an error is detected (output `Error` = TRUE). The parameter `PAR.EnRV` must activate the usage of the value `RV`.

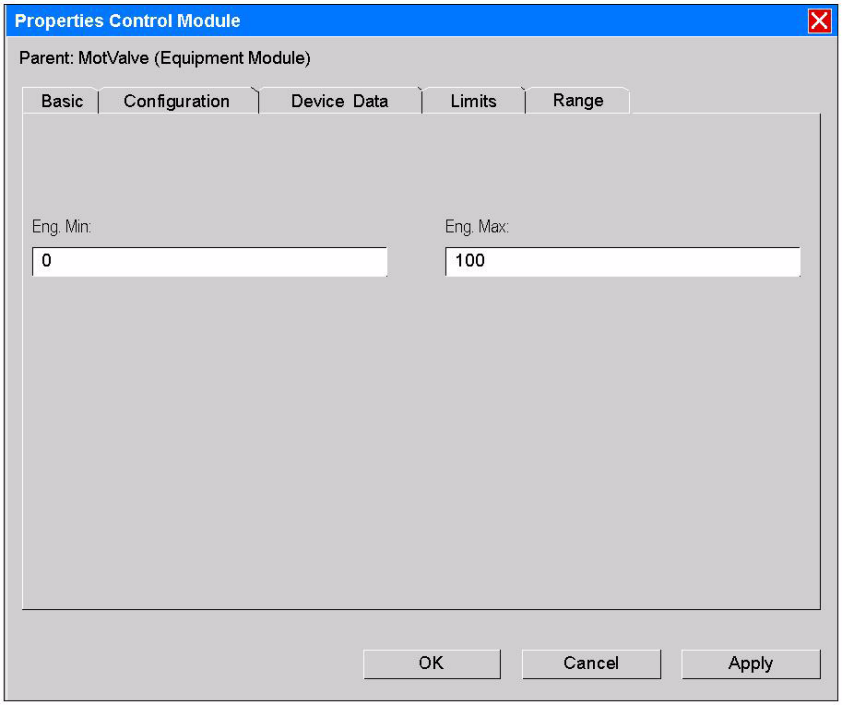
Eng. Min/Eng. Max Property of the Control Module

Overview

The **Eng. Min** and **Eng. Max** property fields of the **Range** tab in the **Properties Control Module** window are used for defining the minimum and maximum values for the control module.

The minimum and maximum values are inherited in the SCoD as scaling boundaries for all configured variables. These values are entered as scaling minimum and maximum limits inside the Vijeo Citect. They are used for limiting user entries in the HMI in trends and in bar graphs.

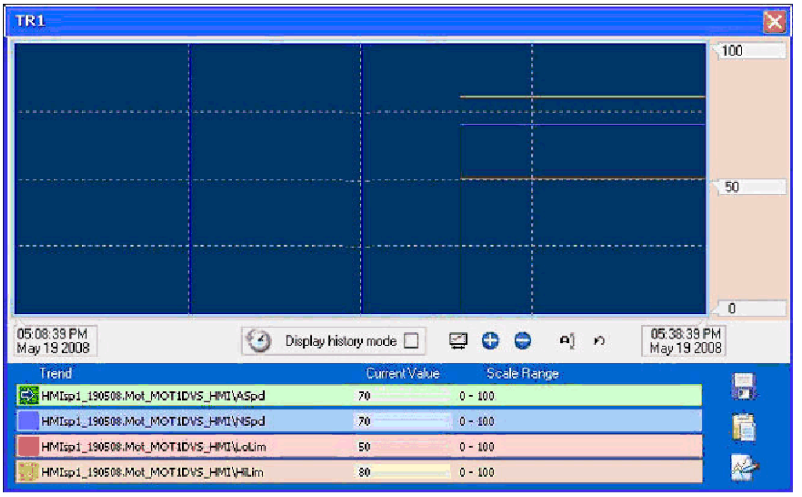
The following figure shows an example of the **Range** tab for the `Mot1dVS` block:



In this example **Eng. Min** and **Eng. Max** are preconfigured as 0 and 100.

These values are inherited in SCoD as an **Eng. Min** and **Eng. Max** for the variables `HMI.LoLim`, `HMI.HiLim`, `HMI.NSpd` and `HMI.Aspd`. This ensures that the user must not enter values beyond the preconfigured values (< `Eng. Min` = 0 and > `Eng. Max` = 100).

The following figure shows the example of a Trend graph for the `Mot1dVS` block having predefined scaling minimum and scaling maximum values as 0 and 100 respectively:



If these analog variables are not preconfigured with the values 0 and 100, the default values are considered as 0 and 32000.

Instantiation in UAG

7

Overview

This chapter gives a general overview about the UAG customization and the UAG generator.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
UAG Customization	92
UAG Generator	94

UAG Customization

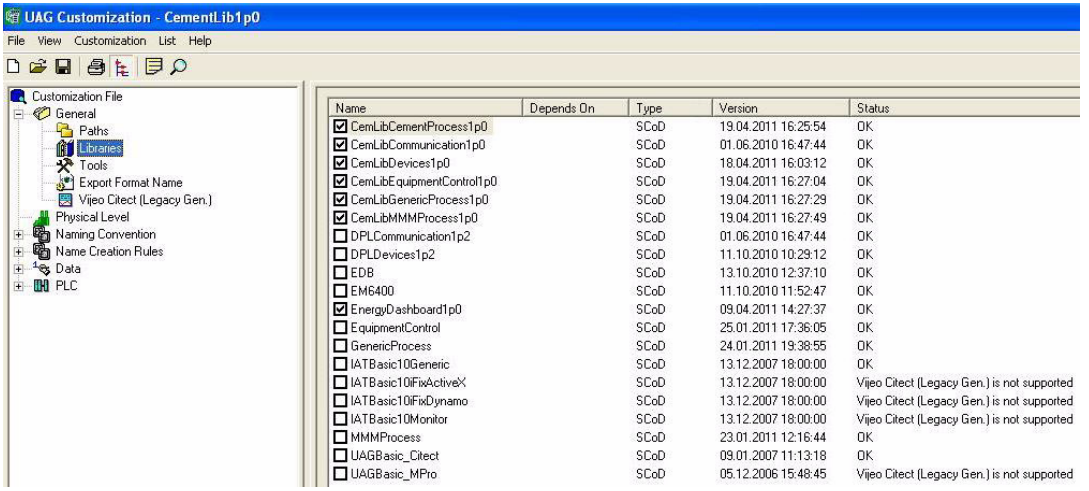
General

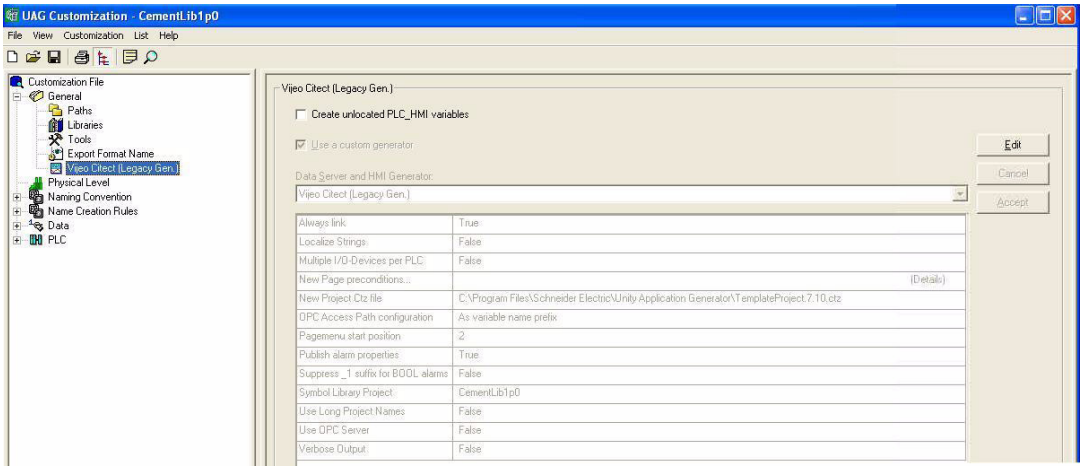
The Cement Library V1.1 enables to generate DFB and HMI Genies for the selected control modules in UAG.

The customization file in the UAG must be configured with Vijeo Citect as HMI. For example, CementLib1P0 (CemLibCementProcess1p0, CemLibCommunication1p0, CemLibDevices1p0, CemLibEquipmentControl1p0, CemLibGenericProcess1p0 and CemLibMMMProcess1p0) are the selected library. Further, select the appropriate Communication Protocol Modnet or OPC Server.

Also the Symbol Library Project (Vijeo Citect Library project) is, for example, set to CementLib1p0 (see the following figure).

A default customization of CementLib1p0.osc is provided with the library (n.n = version).





By default, UAG Customization PLC HMI Communication is set for modnet communication protocol. The OPC server for PLC HMI communication is activated by clicking **Edit** and then changing the **Use OPC Server** from **False** to **True** (see the figure).

UAG Generator

General

The required physical model and topological model must be defined in the UAG.

The property pages of the control module are the supporting pages for Cement Library features.

The properties of a control module are used for configuring the following:

- initial values
- additional data

The properties of a control module can be configured and modified for each configured parameter.

A new control module from the Cement Library can be added by right-clicking the control module container level (Equipment level) and selecting the appropriate control module from the **Control Module Type**.

For detailed information, refer to the *Unity Application Generator User Manuals*.

⚠ WARNING

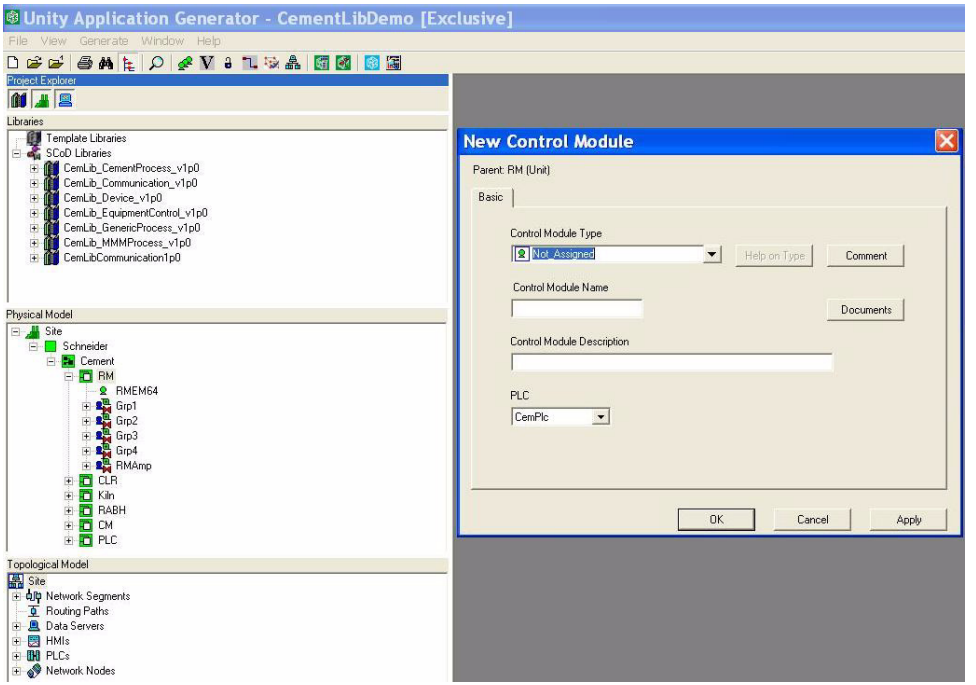
RISK OF PROCESSING INCORRECT DATA

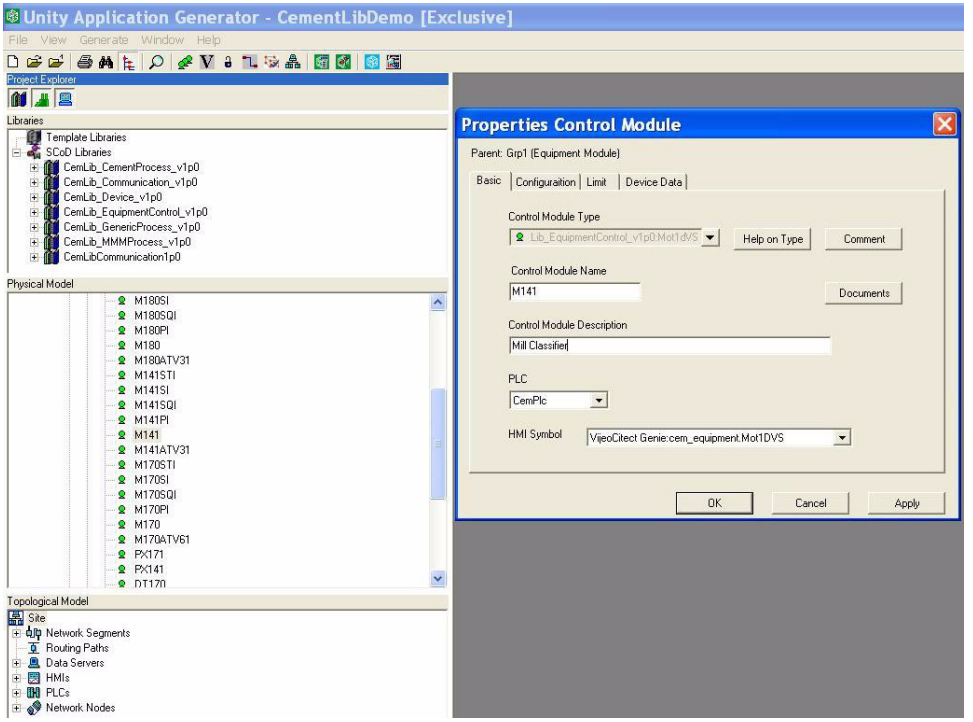
Changes during runtime are not updated in the UAG project. The control engineer must update changes in the environment.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

Step 1: Adding a Control Module

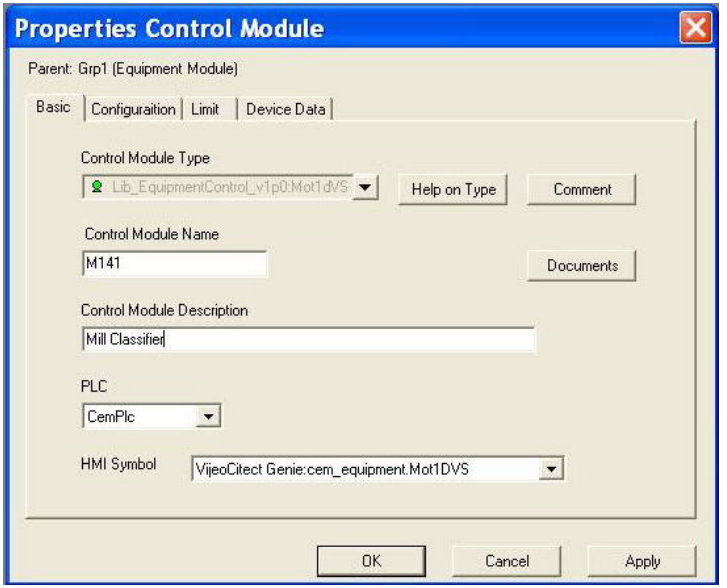
To add a control module in the Unity Application Generator:

Step	Action
1	Select a **New Control Module** 

Step	Action
2	Select a type from **Control Module Type**.
3	Click **Ok**. **Result:** The **Properties Control Module** dialog box is displayed. 

Step 2: Properties of a Control Module - Basic Tab

The following figure displays the **Basic** tab for setting the control module name, description, and PLC.



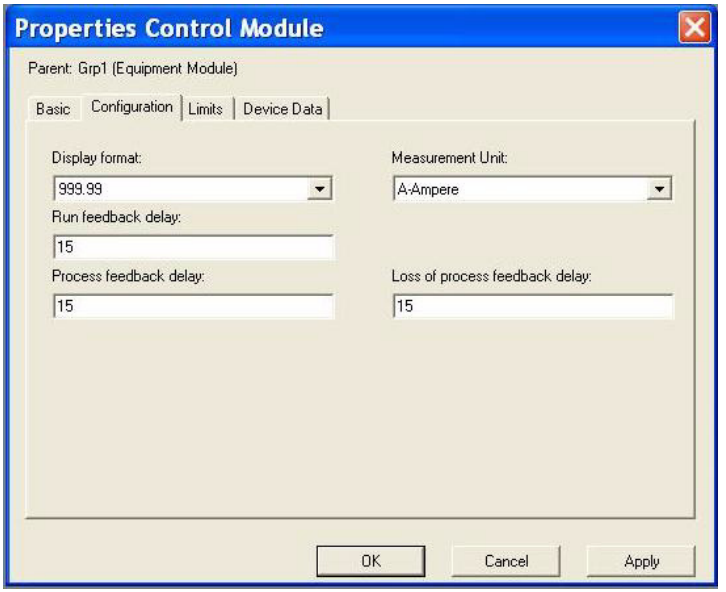
This enables you to set the following fields:

- **Control Module Name**
- **Control Module Description**
- **PLC**

Step 3: Properties of a Control Module - Configuration Tab

All the SCoD properties defined for a control module can be set and modified from 1 or more tabs of the **Properties Control Module** dialog box.

The following figure shows an example of the **Properties Control Module** dialog box with **Configuration** tab.

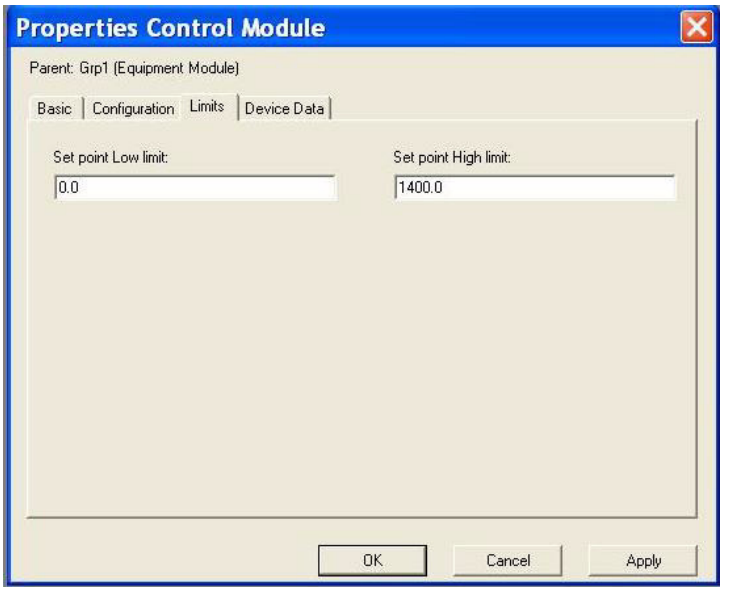


This tab enables to configure the following fields:

- **Display format**
- **Measurement Unit**
- **Run feedback delay**
- **Process feedback delay**
- **Loss of process feedback delay**

Step 4: Properties of a Control Module - Limits Tab

Each control module contains a **Limits** tab to configure the information related to the control module as shown in the following figure.

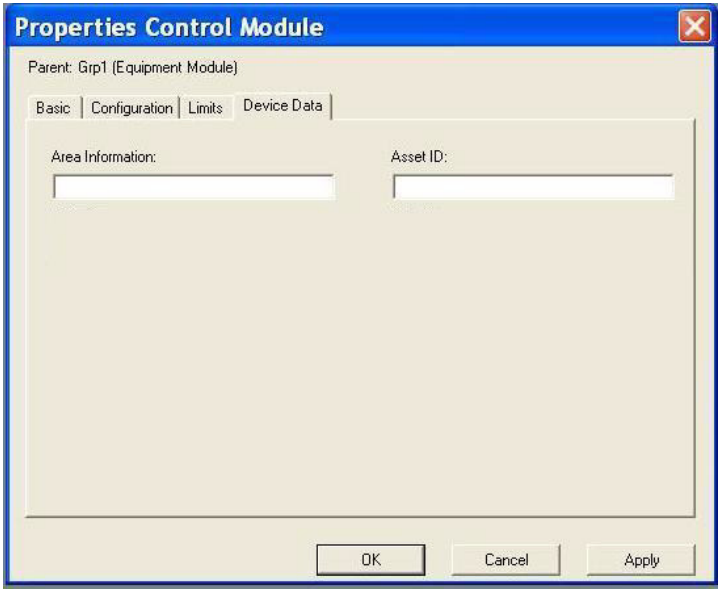


This tab enables to configure the following fields:

- **Set point Low limit**
- **Set point High limit**

Step 5: Properties of a Control Module - Device Data Tab

Each control module contains a **Device Data** tab to configure the information related to the control module as displayed in the following figure.

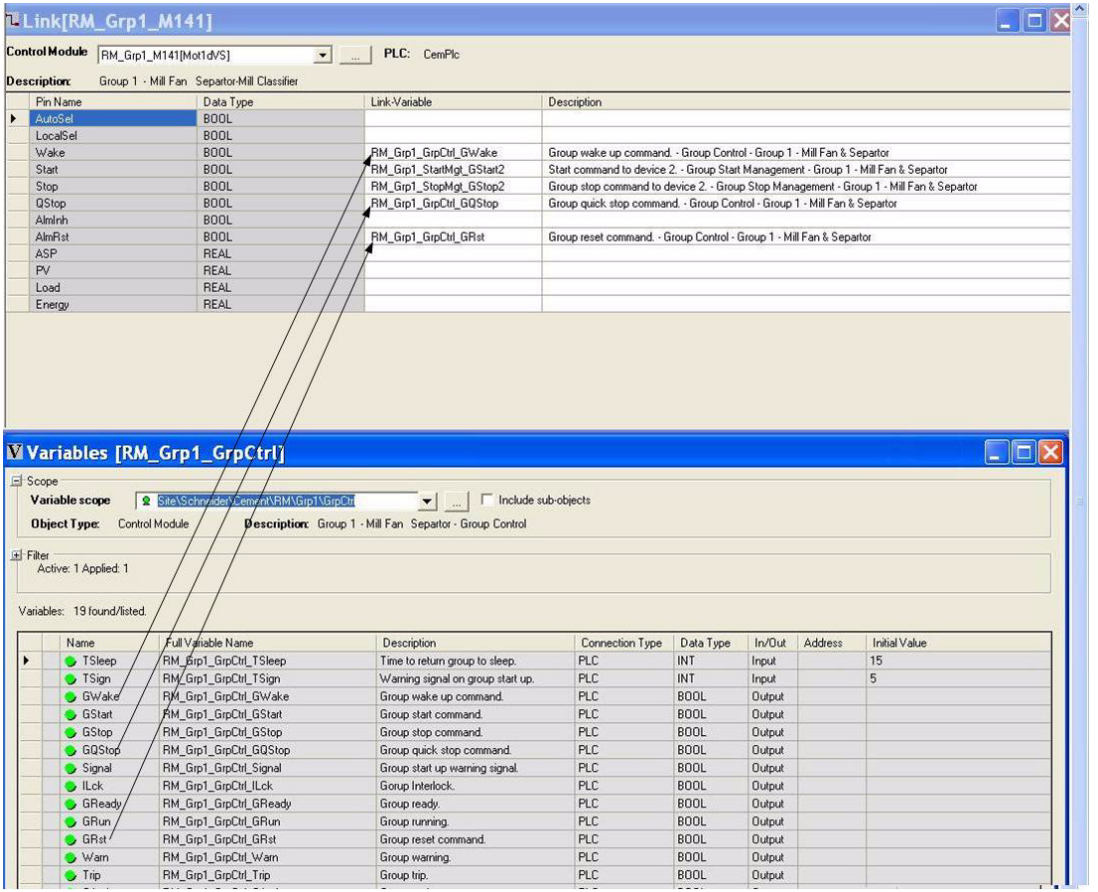


This tab enables to configure the following additional information used by the operator and the maintenance department:

- **Area Information**
- **Asset ID**

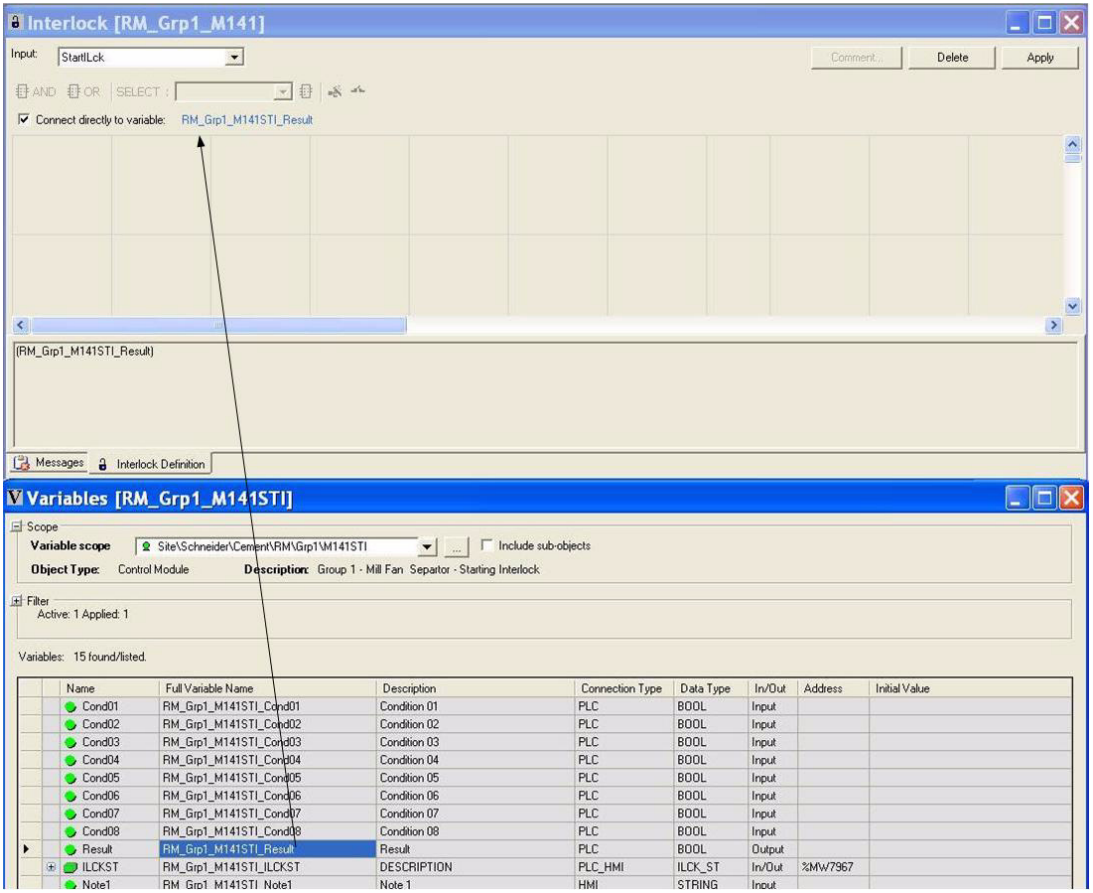
Step 6: Configuring Links for a Control Module

The links for each control module must be connected with suitable variables by dragging and dropping the variable as displayed in the following figure.



Step 7: Configuring Interlocks for a Control Module

The control modules types of motors and actuators must be defined with interlock pin and with suitable interlock definition as shown in the following figure.



Name	Full Variable Name	Description	Connection Type	Data Type	In/Out	Address	Initial Value
Cond01	RM_Grp1_M141STI_Cond01	Condition 01	PLC	BOOL	Input		
Cond02	RM_Grp1_M141STI_Cond02	Condition 02	PLC	BOOL	Input		
Cond03	RM_Grp1_M141STI_Cond03	Condition 03	PLC	BOOL	Input		
Cond04	RM_Grp1_M141STI_Cond04	Condition 04	PLC	BOOL	Input		
Cond05	RM_Grp1_M141STI_Cond05	Condition 05	PLC	BOOL	Input		
Cond06	RM_Grp1_M141STI_Cond06	Condition 06	PLC	BOOL	Input		
Cond07	RM_Grp1_M141STI_Cond07	Condition 07	PLC	BOOL	Input		
Cond08	RM_Grp1_M141STI_Cond08	Condition 08	PLC	BOOL	Input		
Result	RM_Grp1_M141STI_Result	Result	PLC	BOOL	Output		
ILCKST	RM_Grp1_M141STI_ILCKST	DESCRIPTION	PLC_HMI	ILCK_ST	In/Out	%MW7967	
Note1	RM_Grp1_M141STI_Note1	Note 1	HMI	STRING	Input		

Object Description

II

Overview

This part provides detailed information on the function blocks.

What's in this Part?

This part contains the following chapters:

Chapter	Chapter Name	Page
8	Equipment Control	105
9	MMM Process	335
10	Cement Process	461
11	Generic Process	531

Equipment Control

8

Overview

The following chapter describes the equipment control objects.

What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
8.1	`Mot1d`: Motor 1 Direction	106
8.2	`Mot2d`: Motor 2 Direction	133
8.3	`AnlgIn`: Analog Input	161
8.4	`AnlgOut`: Analog Output	174
8.5	`DigIn`: Digital Input	185
8.6	`DigOut`: Digital Output	198
8.7	`Mot1dVS`: Motor 1 Direction Variable Speed	207
8.8	`DigAct`: Digital Actuator	236
8.9	`MotAct`: Motor Actuator	260
8.10	`AnlgAct`: Analog Actuator	287
8.11	`MotBasic`: Motor Basic	316

8.1 Mot1d: Motor 1 Direction

Overview

This section describes the Mot1d block.

What's in this Section?

This section contains the following topics:

Topic	Page
Mot1d - Description	107
Mot1d - Representation as SCoD in UAG	119
Mot1d - Representation in the HMI	122
Mot1d_St Derived Data Type	130

Mot1d - Description

General

The Mot1d block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB Mot1d
Vijeo Citect	Genie Mot1d
UAG	SCoD Mot1d

Function Description

The Mot1d block is used to control electrical motors with 1 direction of rotation.

This block supports 3 modes of operation:
- Auto mode
- Manual mode
- Local mode

By default, the motor will be in Auto mode. The motor will be forced to Auto or Local mode if the mode selection input pin (AutoSel or LocalSel) is HIGH. Local pin has the highest priority. When the Local mode is selected, the block functions in Local mode irrespective of other mode selections.

If the mode selection input pin (AutoSel or LocalSel) is LOW, operator changes the mode of motor from the popup of motor block. When the motor is under maintenance, the Out of Service mode can be activated from the popup.

Auto Mode Operation

A wake-up command (**Wake**) is given by the GrpCtrl block to wake-up the motor from sleep. A group start-up pulse command (**Start**) is given either from the GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group. This sets the output GActive as HIGH if all the interlocks (StartILck, SftyILck, SeqILck, ProcILck) and all the feedbacks (Lstop, Avail, OverLoad, EStop) are HIGH. This sets the output (OP) as HIGH when the input conditional start (CondStart) is HIGH, and motor starts in the selected direction.

Ready output (Ready) is set HIGH when all interlocks and conditions are HIGH.

Interlock output (ILck) is set HIGH when inputs starting interlock (StartILck), safety interlock (SftyILck), sequential interlock (SeqILck) and process interlock (ProcILck) are set HIGH.

After the start command (**Start**) is issued and output (OP) is set HIGH within the specified run feedback delay, the electrical run feedback (FbRun) has to be obtained. This electrical run feedback is continuously monitored by a motor block. If it is not operational, motor is tripped, and an alarm is generated.

After the start command (**Start**) is issued and motor is operational, if run feedback (FbRun) is not available for 500 ms, the motor is tripped, and an alarm is generated.

After the start command (**Start**) is issued and the output (OP) is set HIGH within the specified process feedback delay, the process feedback (FbProc) has to be obtained. This process feedback is continuously monitored by the motor block. If it is not operational, motor is tripped, and an alarm is generated.

After the start command (**Start**) is issued, output (OP) is set HIGH, and motor is operational. If process feedback (FbProc) is not available for specified loss of process feedback delay, the motor is tripped, and an alarm is generated.

NOTE: The process feedback can be bypassed by selecting the **Bypass Process f/b** button from the HMI.

A conditional start can be given to this block when the Conditional Start is enabled. This input is used for connecting process logic to control the motor under normal conditions.

NOTE: Conditional Start, Group Active, Sequential Start and Sequential Stop are applicable in Auto mode only.

The start command (**Start**) is issued, and output (OP) is set HIGH when run feedback (FbRun) and process feedback (FbProc) are obtained. Run output (Run) is set HIGH, and pulse sequential stop command (**SeqStart**) is given to the next device to start.

Group stop command (**Stop**) or group quick stop command (**QStop**) sets the output command (**OP**) as LOW. Group stop command (**Stop**) is received by the motor from the GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group. Group quick stop command (**QStop**) is received by the motor from the GrpCtrl block.

The stop command (**Stop**) is issued and output (OP) is set LOW when the run feedback (FbRun) and process feedback (FbProc) are set LOW. Run output (Run) is set LOW and sequential stop command (**SeqStop**) is given to the next device to stop.

The quick stop command (**QStop**) is issued and output (OP) is set LOW, and sequential stop command (**SeqStop**) is given to the next device to stop.

Manual Mode Operation

Run and **Stop** commands are issued by the operator from the popup of the motor block. Run and process feedback monitoring are applicable as in Auto mode.

NOTE: Sequential interlock is not applicable in Manual mode.

Local Mode Operation

Stage	Description
1	Start-up pulse command (**LStart**) comes from the local control panel of the motor that sets the output `OP` HIGH and motor starts in the selected direction.
2	Stop command (**LStop**) also comes from the local control panel of the motor. This input always must be HIGH for the motor to be started and is applicable in all the modes of operation. **NOTE:** This serves as an emergency stop command.

Run and process feedback monitoring are applicable as in Auto mode.

Interlock Types

There are 4 types of interlocks. The result of individual interlock condition blocks is connected to these interlock pins:

Interlock Name	Parameter	Description
Starting Interlock	`StartILck`	These interlocks are valid before motor start-up only.
Safety Interlock	`SftyILck`	All the safety conditions are taken into consideration. This input is active in all the modes.
Sequential Interlock	`SeqILck`	All the sequential conditions are taken into consideration. This input is active only in Auto mode.
Process Interlock	`ProcILck`	All the process conditions are taken into consideration. This input is active in Auto and Manual mode.

Feedback

The status of the interlock is displayed in the popup. Detailed description of the interlocks is displayed in the **Interlock** tab of the popup.

Local stop signal (`LStop`) is connected from stop push button of local panel and is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated in Auto and Manual modes only.

The availability (`Avail`) of the motor is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated. The overload feedback (`OverLoad`) of the motor is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated.

An emergency stop signal (`EStop`) comes from MCC panel emergency shut-down push button and is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated.

The alert and trip are indicated in the SCADA. The number of active alarms is shown in the popup of the motor. The details of the active alarms are shown in the **Alarm** tab of the popup. Based on configured conditions, the alarms for this block can be inhibited by connecting the logic to the input pin `AlmInh`. The input pin `AlmRst` that is connected from `GrpCtrl` block or the operator from popup can reset the trip alarms.

Configuration

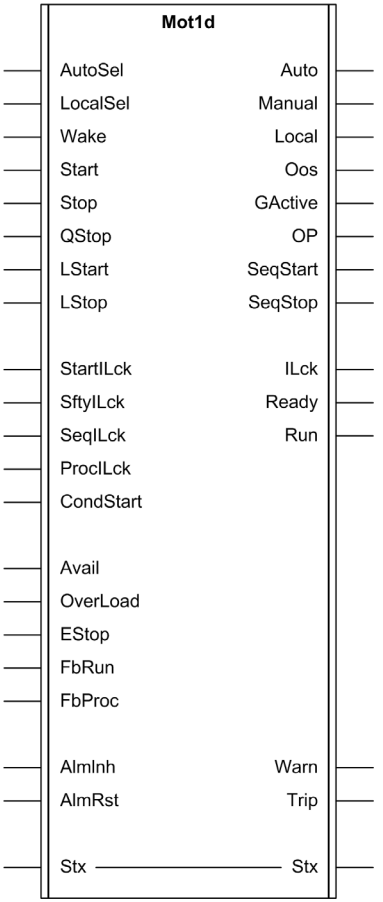
The run hours of the motor is displayed in the **Configuration** tab of the popup. After the maintenance of the motor, the run hours can be reset from the popup.

NOTE: The following parameters in the **Configuration** tab have user restriction from the HMI:

- **Run feedback delay**
- **Process feedback delay**
- **Loss of Process feedback delay**
- **Run hours R** (reset)

Representation in FBD

The following figure shows the FBD representation of the `Mot1d` block.



Parameter Description

The following table describes the input parameters of the Mot1d block:

Parameter	Type	Description	Details
AutoSel	BOOL	Auto mode selection	Selection of operation mode (Auto mode): • AutoSel = 1: Block is forced to Auto mode. Mode selection option in HMI is grayed out. • AutoSel = 0: Block is set free for operator to select the mode from HMI.
LocalSel	BOOL	Local mode selection	Selection of operation mode (Local mode): • LocalSel = 1: Block is forced to Local mode. Mode selection option in HMI is grayed out. • LocalSel = 0: Block is set free for operator to select the mode from HMI.
Wake	BOOL	Group wake-up	Group wake-up and request for group availability check: • Wake = 1: Group is active. • Wake = 0: Group is in Sleep mode. This input is connected from output GWake of GrpCtrl.
Start	BOOL	Group start	**Start** command to the motor. Start = 1: OP pin is set to 1, this is applicable in Auto mode only. This input is connected from the GStart pin of GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group.
Stop	BOOL	Group stop	**Stop** command to the motor. Stop = 1: OP pin is set to 0, this is applicable in Auto mode only. This input is connected from the GStop pin of GrpCtrl or GrpStpMgt or RouteMgt or previous device in the group.
QStop	BOOL	Group quick stop	**Stop** command to the motor. QStop = 1: OP pin is set to 0, this is applicable in Auto mode only. This input is connected from the GQStop pin of GrpCtrl.

Parameter	Type	Description	Details
LStart	BOOL	Local start	**Start** command to the motor. LStart = 1: OP pin is set to 1, this is applicable in Local mode only. This input is connected from the **Start** push button of local IO panel.
LStop	BOOL	Local stop	**Stop** command to the motor. LStop = 0: OP pin is set to 0, this is applicable in Local mode only. This input is connected from the **Stop** push button of local IO panel.
StartILck	BOOL	Starting interlock	• StartILck = 1: The starting interlock conditions are operational. • StartILck = 0: The starting interlock conditions are not operational. Applicable only in Auto and Manual modes of operation. Applicable only before starting of the motor. After motor is started, StartILck is not considered.
SftyILck	BOOL	Safety interlock	• SftyILck = 1: The safety interlock conditions are operational. • SftyILck = 0: The safety interlock conditions are not operational, trips the motor and generates an alarm. Applicable in all the modes of operation (Auto, Manual, and Local).
SeqILck	BOOL	Sequential interlock	• SeqILck = 1: The sequential interlock conditions are operational. • SeqILck = 0: The sequential interlock conditions are not operational, trips the motor and generates an alarm. Applicable in Auto mode only.

Parameter	Type	Description	Details
ProcILck	BOOL	Process interlock	• ProcILck = 1: The process interlock conditions are operational. • ProcILck = 0: The process interlock conditions are not operational, trips the motor and generates an alarm. Applicable in Auto and Manual modes only.
CondStart	BOOL	Conditional start	• CondStart = 1: **Start** command to the motor, OP = 1. • CondStart = 0: **Stop** command to the motor, OP = 0. Applicable in Auto mode only. If GActive = 1, CondStart is used to start or stop a motor based on process condition.
Avail	BOOL	Availability of motor	• Avail = 1: Motor is electrically available. • Avail = 0: Motor is electrically not available. **Stop** command to the motor, OP = 0. This signal comes from the MCC outgoing circuit (electrical protection active, circuit breaker switched off, drawer out, and so on).
OverLoad	BOOL	Overload	• OverLoad = 1: Motor is not in overload condition. • OverLoad = 0: Motor is in overload condition. **Stop** command to the motor, OP = 0. This signal comes from the MCC outgoing circuit (electrical protection active).

Parameter	Type	Description	Details
EStop	BOOL	Emergency stop	• EStop = 1: Plant process condition is operational for the motor to run. • EStop = 0: Emergency situation. Plant process condition is not operational for the motor to run. **Stop** command to the motor, OP = 0. Applicable in all operation modes (Auto, Manual, and Local). This input is connected from the **Emergency Stop** push button in the field.
FbRun	BOOL	Run feedback of motor - Electrical	Once the motor is given a **Start** command from output OP, its electrical run feedback is continuously monitored. This signal comes from the MCC outgoing circuit. If OP = 1 and FbRun = 0, then **Stop** command to the motor, OP = 0 is given, and an alarm is generated after time delay. If OP = 0 and FbRun = 1, an alarm is generated after time delay.
FbProc	BOOL	Run feedback of motor - Process	Once the motor is given a **Start** command from output OP, its electrical run feedback is continuously monitored. This signal comes from Speed monitoring devices, Zero Speed switch, Proximity switches, and so on. If OP = 1 and FbProc = 0 then **Stop** command to the motor, OP = 0 is given and an alarm is generated after time delay. If OP = 0 and FbProc = 1, an alarm is generated after time delay.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

The AlmInh parameter to inhibit alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the Mot1d block:

Parameter	Type	Description	Details
AlmInh	BOOL	Alarm inhibition	• AlmInh = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • AlmInh = 0: Alarms are not inhibited.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

To reset, alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the Mot1d block:

Parameter	Type	Description	Details
AlmRst	BOOL	Reset	• AlmRst = 1: Resets all the alarms generated by the block. • AlmRst = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from GRst pin of GrpCtrl block and resets the trip alarm.

The following table describes the output parameters of the Mot1d block:

Parameter	Type	Description	Details
Auto	BOOL	Auto mode	Auto = 1: Auto mode is selected. Motor operation is group based on GStart and GStop.
Manual	BOOL	Manual mode	Manual = 1: Manual mode is selected. Motor operation is based on commands **Start** and **Stop** given by operator from HMI.
Local	BOOL	Local mode	Local = 1: Local mode is selected. Motor operation is based on commands **LStart** and **LStop** given from field.

Parameter	Type	Description	Details
Oos	BOOL	Out of Service mode	Oos = 1: Out of Service mode is selected. Motor is not operated and OP = 0.
GActive	BOOL	Group active	The motor is started in the group and is active in the group. Start command pulse **Start** is obtained, GActive = 1.
OP	BOOL	Output command to motor.	• OP = 1: Motor is given command to start. • OP = 0: Motor is given command to stop. The output switches on the contactor or circuit breaker controlling the motor in forward direction.
SeqStart	BOOL	Start command to next motor in group.	Start command (pulse) is received and outputs OP and Run = 1, sequential start command pulse is given to the next motor connected in the group.
SeqStop	BOOL	Stop command to next motor in group.	Stop or QStop = 1 and OP and Run = 0, sequential stop command is given to the next motor connected in the group.
ILck	BOOL	Interlock status	• ILck = 1: All the interlocks are operational. • ILck = 0: All the interlocks are not operational. There is an active interlock present.
Ready	BOOL	Motor ready to start.	• Ready = 1: All the conditions to start the motor are operational. • Ready = 0: The conditions to start the motor are not operational. Motor will not be given **Start** command.
Run	BOOL	Motor running	Motor is running. This output is given if the motor is running and run feedback (FbRun) is received. Applicable in all the operation modes.

Parameter	Type	Description	Details
Warn	BOOL	Alert	• Warn = 1: The motor is operational with no active alert present. • Warn = 0: The motor is operational with an active alert present.
Trip	BOOL	Trip	• Trip = 1: The motor is operational with no active trip alarm present. • Trip = 0: The motor has tripped with an active trip alarm present.

The following table describes the input/output parameter of the Mot1d block:

Parameter	Type	Description
Stx	Mot1d_St	Status and control of the motor in 1 direction. Refer to the Structure to interface with the HMI *(see page 130)*.

Mot1d - Representation as SCoD in UAG

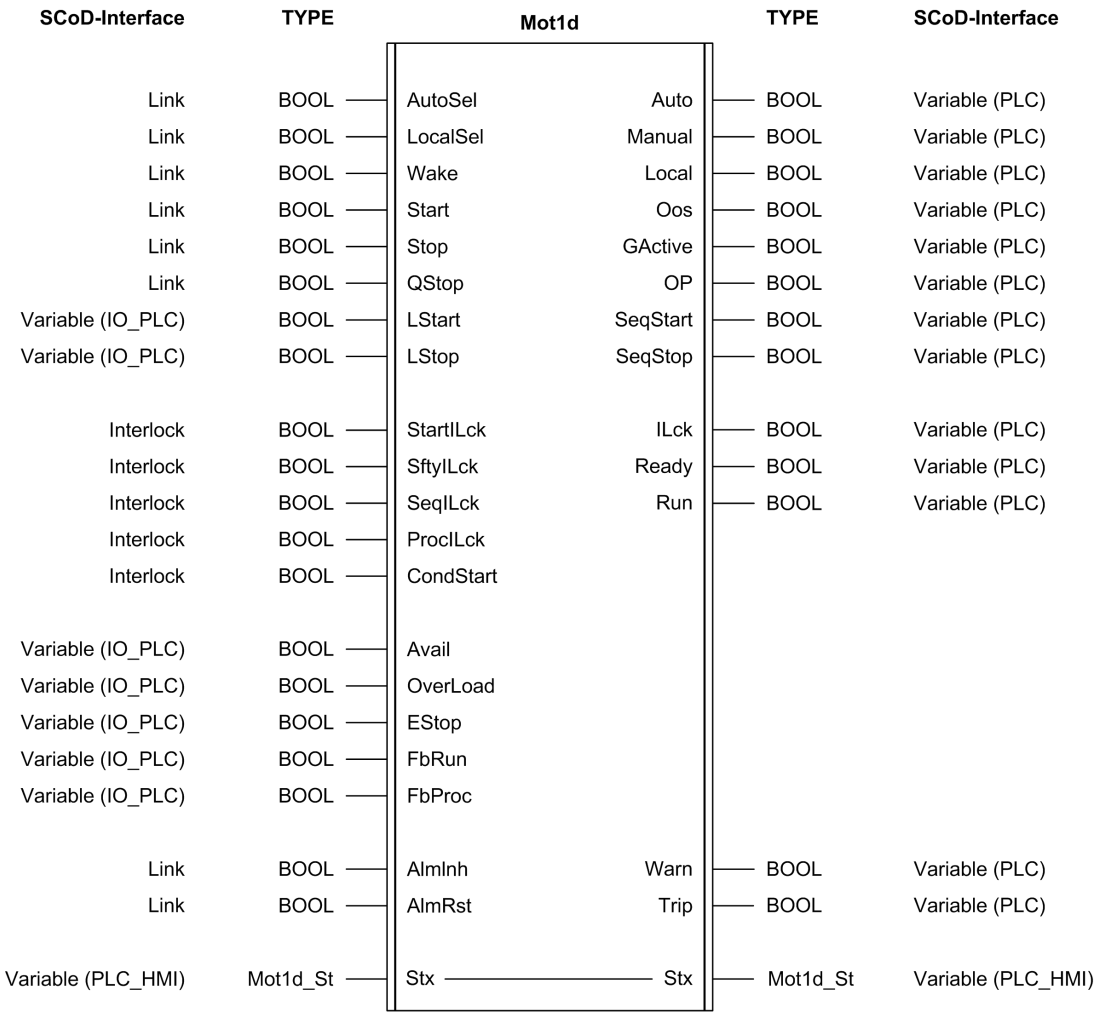
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the Stx variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type HMI is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the `Mot1d` block.



SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Area Information	`AreaInfo`	Not available	**Device Data**	1	Area information
Asset ID	`AssetId`	Not available	**Device Data**	7	Asset ID

Name	Variable	Initial Value	Property Tab	Position	Description
Display Format (time)	`Stx.RunHr`	#####	**Configuration**	1	Display format
Run fb delay (sec)	`Stx.TRunFb`	0	**Configuration**	2	Sets run feedback delay (s).
Process feedback delay (sec)	`St.TProcFb`	0	**Configuration**	8	Sets process feedback delay (s).
Loss of process fb delay	`St.TLProcFb`	0	**Configuration**	3	Sets loss of process feedback delay (s).

Mot1d - Representation in the HMI

Motor 1 Direction (Genie)

The Mot1d block is represented in the HMI by the following genie.

Genie Elements

Symbol	Meaning
	Opens the Mot1d popup.
	The instance name of the Mot1d block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

Animation Description of the Genie

Symbol	Animation	Meaning
	Line color: Steady light gray Fill color: Steady light gray	Motor is in Sleep mode.
	Line color: Steady dark gray Fill color: Steady white	• Motor is awake or in Local mode or Manual mode and ready. • Motor is awake or in Local mode or Manual mode and ready after alarms are reset.
	Line color: Dark gray Fill color: Light gray	Motor is awake or in Local mode or Manual mode and not ready.
	Line color: Steady dark gray Fill color: Flashing green	Motor is started and feedback is yet to be obtained or motor is stopping.
	Line color: Steady dark gray Fill color: Steady green	Motor is running.
	Line color: Flashing yellow Fill color: Steady green	Motor is running and an alert alarm is active.
	Line color: Steady yellow Fill color: Steady green	Motor is running and an alert alarm is acknowledged.

Symbol	Animation	Meaning
	Line color: Steady dark gray Fill color: Flashing red	Motor is tripped and a trip alarm is active.
	Line color: Steady dark gray Fill color: Steady red	Motor is tripped and a trip alarm is acknowledged.

Modes

On the top left corner of the motor 1 direction genie, the mode of operation is indicated.

The following table describes the mode of operations:

Mode	Meaning
A	Motor is operating in the Auto mode.
M	Motor is operating in the Manual mode.
L	Motor is operating in the Local mode.
O	Motor is in Out of Service mode.

Item	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Alarm Masking

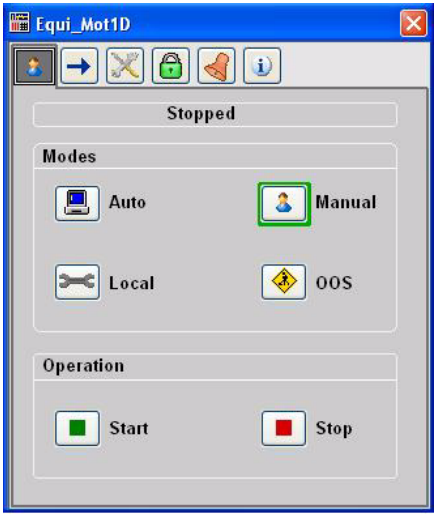
Symbol	Meaning
	Trip or alert alarms in the motor 1 direction are masked.

Popup

The `Mot1d` block popup has the following tabs:

- **Operator** tab
- **Extended** tab
- **Configuration** tab
- **Interlock** tab
- **Alarm** tab
- **Information** tab

Operator Tab



This tab has 3 sub-sections:

- **Status**
- **Modes**
- **Operation**

The sub-section **Status** displays the current status.

The possible displays are listed as follows:

- **Sleeping**
- **Group Active**
- **Ready**
- **Not Ready**
- **Starting**
- **Stopping**
- **Running**
- **Interlocked**
- **Warning**
- **Trip**

The following table describes the sub-section **Modes**:

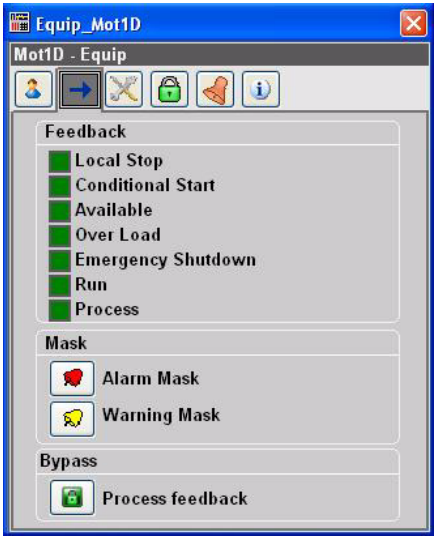
Item	Description
Auto	Switches to Auto operation mode controlled by the PLC program.
Manual	Switches to Manual operation mode controlled by the HMI.

Item	Description
Local	Switches to Local operation mode controlled by the local control box of the motor.
OOS	Switches to Out of Service mode for maintenance purpose of the object (applicable in all modes).

The following table describes the sub-section **Operation**:

Item	Description
Start	Initiates the operation of the device in Manual mode.
Stop	Stops the operation of the device in Manual mode.

Extended Tab



This tab has 3 sub-sections:

- **Feedback**
- **Mask**
- **Bypass**

The following table describes the sub-section **Feedback**:

Item	Animation	Description
Local stop	Steady green	Motor locally stopped feedback is obtained.
	Steady gray	Motor locally not stopped feedback is obtained.

Item	Animation	Description
Conditional Start	Steady green	Motor is started.
	Steady gray	Motor is not started.
Available	Steady green	Motor electrically available feedback is obtained.
	Steady gray	Motor electrically not available feedback is obtained.
Over Load	Steady green	Motor overload feedback is obtained.
	Steady gray	Motor overload feedback is not obtained.
Emergency shutdown	Steady green	Emergency stop feedback is obtained.
	Steady gray	Emergency stop feedback is not obtained.
Run	Steady green	Motor running feedback is obtained.
	Steady gray	Motor not running feedback is obtained.
Process	Steady green	Process feedback is obtained.
	Steady gray	Process feedback is not obtained.

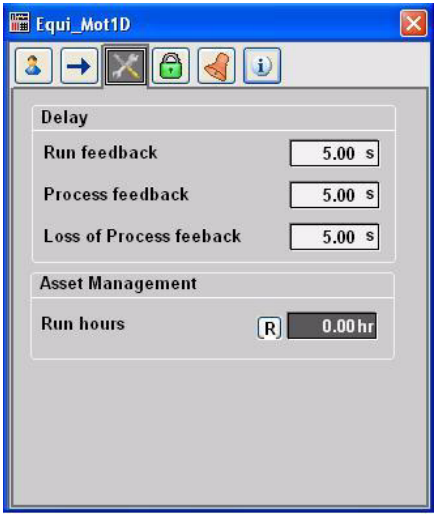
The following table describes the sub-section **Mask**:

Item	Description
Alarm Mask	Masks the indicated current trip alarms of this block in the alarm list.
Warning Mask	Masks the current alerts of this block in the alarm list.

The following table describes the sub-section **Bypass**:

Item	Description
Process feedback	Bypasses the process feedback

Configuration Tab



This tab has 2 sub-sections:

- **Delay**
- **Asset Management**

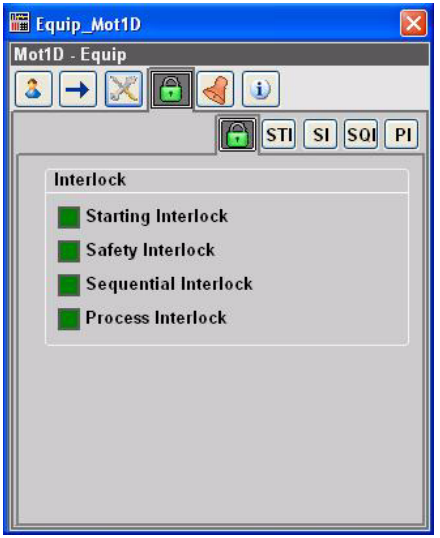
The following table describes the sub-section **Delay**:

Item	Description
Run feedback	Displays run feedback delay of motor, configurable from HMI. After starting motor, if running feedback is not received before delay time, an alarm is generated, and the motor is tripped.
Process feedback	Displays process feedback delay of motor, configurable from HMI. After starting motor, if process feedback is not received before delay time, an alarm is generated, and the motor is tripped.
Loss of Process feedback	Displays loss of process feedback delay of motor, configurable from HMI. After receiving process feedback, if there is loss of feedback while motor running and is not received before delay time, an alarm is generated, and the motor is tripped.

The following table describes the sub-section **Asset Management**:

Item	Description
Run hours	Displays the run hours of the motor.
	Resets the run hours value of the motor to 0.

Interlock Tab



This tab has a sub-section called as **Interlock**, which is described in the following table:

Item	Animation	Description
Starting Interlock	Steady green	**Starting Interlock** is active, and the related device is not blocked.
	Steady gray	**Starting Interlock** is not active, and the related device is blocked.
Safety Interlock	Steady green	**Safety Interlock** is active, and the related device is not blocked.
	Steady gray	**Safety Interlock** is not active, and the related device is blocked.

Item	Animation	Description
Sequential Interlock	Steady green	**Sequential Interlock** is active, and the related device is not blocked.
	Steady gray	**Sequential Interlock** is not active, and the related device is blocked.
Process Interlock	Steady green	**Process Interlock** is active, and the related device is not blocked.
	Steady gray	**Process Interlock** is not active, and the related device is blocked.

This tab has sub-tabs Interlock, **STI**, **SI**, **SQI**, and **PI**.

For description about **Interlock** sub-tabs, refer to Commonly Used Tabs *(see page 53)*.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

Mot1d_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type Mot1d_St for the Mot1d block:

Element	Type	Description
RunHr	INT	Run hours of motor
TRunFb	INT	Run feedback delay
TProcFb	INT	Process feedback delay
TLProcFb	INT	Loss of process feedback delay
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word1 to interact with HMI
Stw2	WORD	Status word2 to interact with HMI
Alw1	WORD	Alarm word1 to interact with HMI

Bit description for the Ctw element:

Bit	Description
0	Auto mode selection from HMI
1	Manual mode selection from HMI
2	Local mode selection from HMI
3	Out of Service mode selection from HMI
4	Start command from HMI
5	Not used
6	Stop command from HMI
7	Alarm reset from HMI
8	Run hour reset from HMI
9	Bypass process feedback from HMI
10...13	Not used
14	Alert mask
15	Alarm mask

Bit description for the Stw1 element:

Bit	Description
0	Auto mode selection from HMI
1	Local mode selection from HMI

Bit	Description
2	Group wake
3	Local stop
4	Starting interlock
5	Safety interlock
6	Sequential interlock
7	Process interlock
8	Motor electrically available
9	Motor overload
10	Emergency stop
11	Run feedback
12	Not used
13	Process feedback
14	Alarm inhibition
15	Not used

Bit description for the `Stw2` element:

Bit	Description
0	Auto mode active status
1	Manual mode active status
2	Local mode active status
3	Out of Service mode active status
4	Output `OP` active
5	Not used
6	Motor ready status
7	Motor running status
8...9	Not used
10	Conditional start active status
11	Not used
12	Interlock active status
13	Group active status
14	Alert active status
15	Trip active status

Bit description for the `Alw1` element:

Bit	Description
0	Motor locally stop
1	Safety interlock not available
2	Sequential interlock not available
3	Process interlock not available
4	Motor electrically not available
5	Motor overload
6	Motor emergency stop
7	Run feedback not active when motor is running
8	Run feedback active when motor is not running
9...10	Not used
11	Process feedback not active when motor is running
12	Process feedback active when motor is not running
13	Not used
14	Run hours maximum value reached
15	Active trip alarm

8.2 Mot2d: Motor 2 Direction

Overview

This section describes the Mot2d block.

What's in this Section?

This section contains the following topics:

Topic	Page
Mot2d - Description	134
Mot2d - Representation as SCoD in UAG	147
Mot2D - Representation in the HMI	150
Mot2d_St Derived Data Type	158

Mot2d - Description

General

The Mot2d block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB Mot2d
Vijeo Citect	Genie Mot2d
UAG	SCoD Mot2d

Function Description

The Mot2d block is used to control electrical motors with 2 directions of rotation.

This block supports 3 modes of operation:
- Auto mode
- Manual mode
- Local mode

By default, the motor will be in Auto mode. The motor will be forced to Auto or Local mode if the mode selection input pin (AutoSel or LocalSel) is HIGH. Local pin has the highest priority. When the Local mode is selected, the block functions in Local mode irrespective of other mode selections.

If the mode selection input pin (AutoSel or LocalSel) is LOW, operator changes the mode of motor from the popup of motor block. When the motor is under maintenance, the Out of Service mode can be activated from the popup.

Auto Mode Operation

A wake-up command (**Wake**) is given by the GrpCtrl block to wake-up the motor from sleep. A group start-up pulse command (**Start**) is given either from the GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group. This sets the output GActive as HIGH if all the interlocks (StartILck, SftyILck, SeqILck, ProcILck) and all the feedbacks (Lstop, Avail, OverLoad, EStop) are HIGH. This sets the output (OP1) or (OP2) as HIGH when the input conditional start (CondStart) is HIGH, and motor starts in the selected direction. Forward direction has the highest priority.

Ready output (Ready) is set HIGH when all interlocks and conditions are HIGH.

Interlock output (ILck) is set HIGH when inputs starting interlock (StartILck), safety interlock (SftyILck), sequential interlock (SeqILck) and process interlock (ProcILck) are set HIGH.

After the start command (**Start**) is issued and outputs (OP1) and (OP2) are set HIGH within the specified run feedback delay, the electrical run feedback (FbFwd) or (FbRev) has to be obtained. This electrical run feedback is continuously monitored by a motor block. If it is not operational, motor is tripped, and an alarm is generated.

After the start command (**Start**) is issued and motor is operational, if run feedback (FbFwd) or (FbRev) is not available for 500 ms, the motor is tripped, and an alarm is generated.

After the start command (**Start**) is issued and the output (OP1) or (OP2) is set HIGH within the specified process feedback delay, the process feedback (FbProc) has to be obtained. This process feedback is continuously monitored by the motor block. If it is not operational, motor is tripped, and an alarm is generated.

After the start command (**Start**) is issued, output (OP1) or (OP2) is set HIGH, and motor is operational. If process feedback (FbProc) is not available for specified loss of process feedback delay, the motor is tripped, and an alarm is generated.

NOTE: The process feedback can be bypassed by selecting the **Bypass Process f/b** button from the HMI.

Direction change of the motor demands stoppage of the motor and a time delay before restarting the motor in other direction.

A conditional start can be given to this block when the Conditional Start is enabled. This input is used for connecting process logic to control the motor under normal conditions.

NOTE: Conditional Start, Group Active, Sequential Start and Sequential Stop are applicable in Auto mode only.

The start command (**Start**) is issued, and output (OP1) or (OP2) is set HIGH when the run feedback (FbFwd) or (FbRev) and process feedback (FbProc) are obtained. Run output (Run) is set HIGH, and pulse command (**SeqStart**) is given to the next device to start.

Group stop command (**Stop**) or group quick stop command (**QStop**) sets the output (OP1) or (OP2) as LOW. Group stop command (**Stop**) is received by the motor from the GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group. Group quick stop command (**QStop**) is received by the motor from the GrpCtrl block.

The stop command (**Stop**) is issued and output (OP1) or (OP2) is set LOW when the run feedback (FbFwd) or (FbRev) and process feedback (FbProc) are set LOW. Run output (Run) is set LOW and sequential stop command (**SeqStop**) is given to the next device to stop.

The quick stop command (**QStop**) is issued and output (OP) is set LOW and sequential stop command (**SeqStop**) is given to the next device to stop.

Manual Mode Operation

Run Forward, **Run Backward**, and **Stop** commands are issued by the operator from the popup of the motor block. Run and process feedback monitoring are applicable as in Auto mode.

NOTE: Sequential interlock is not applicable in Manual mode.

Local Mode Operation

Stage	Description
1	Start-up in forward direction pulse command (**LStart1**) or start-up in reverse direction pulse command (**LStart2**) comes from the local control panel of the motor that sets the output `OP1` or `OP2` HIGH and motor starts in the selected direction.
2	Stop command (**LStop**) also comes from the local control panel of the motor. This input always must be HIGH for the motor to be started and is applicable in all the modes of operation. **NOTE:** This serves as an emergency stop command.

NOTE: Direction change, run and process feedback monitoring are applicable as in Auto mode.

Interlock Types

There are 4 types of interlocks. The result of individual interlock condition blocks is connected to these interlock pins:

Interlock Name	Parameter	Description
Starting Interlock	`StartILck`	These interlocks are valid before motor start-up only.
Safety Interlock	`SftyILck`	All the safety conditions are taken into consideration. This input is active in all the modes.
Sequential Interlock	`SeqILck`	All the sequential conditions are taken into consideration. This input is active only in Auto mode.
Process Interlock	`ProcILck`	All the process conditions are taken into consideration. This input is active in Auto and Manual mode.

Feedback

The status of the interlock is displayed in the popup. Detailed description of the interlocks is displayed in the **Interlock** tab of the popup.

Local stop signal (`LStop`) is connected from stop push button of local panel and is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated in Auto and Manual modes only.

The availability (`Avail`) of the motor is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated. The overload feedback (`OverLoad`) of the motor is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated.

An emergency stop signal (`EStop`) comes from MCC panel emergency shut-down push button and is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated.

The alert and trip are indicated in the SCADA. The number of active alarms is shown in the popup of the motor. The details of the active alarms are shown in the **Alarm** tab of the popup. Based on configured conditions, the alarms for this block can be inhibited by connecting the logic to the input pin `AlmInh`. The input pin `AlmRst` that is connected from `GrpCtrl` block or the operator from popup can reset the trip alarms.

Configuration

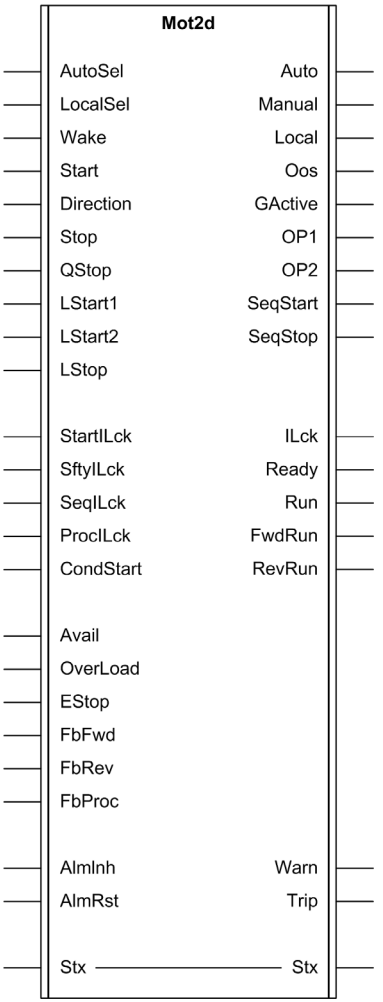
The run hours of the motor is displayed in the **Configuration** tab of the popup. After the maintenance of the motor, the run hours can be reset from the popup.

NOTE: The following parameters in the **Configuration** tab have user restriction from the HMI:

- **Run feedback delay**
- **Process feedback delay**
- **Loss of Process feedback delay**
- **Run hours R** (reset)

Representation in FBD

The following figure shows the FBD representation of the `Mot2d` block.



Parameter Description

The following table describes the input parameters of the Mot2d block:

Parameter	Type	Description	Details
AutoSel	BOOL	Auto mode selection	Selection of operation mode (Auto mode): • AutoSel = 1: Block is forced to Auto mode. Mode selection option in HMI is grayed out. • AutoSel = 0: Block is set free for operator to select the mode from HMI.
LocalSel	BOOL	Local mode selection	Selection of operation mode (Local mode): • LocalSel = 1: Block is forced to Local mode. Mode selection option in HMI is grayed out. • LocalSel = 0: Block is set free for operator to select the mode from HMI.
Wake	BOOL	Group wake-up	Group wake-up and request for group availability check: • Wake = 0: Group is in Sleep mode. • Wake = 1: Group is active. This input is connected from output GWake of GrpCtrl.
Start	BOOL	Group start	**Start** command to the motor. GStart1 = 1: OP pin is set to 1, this is applicable in Auto mode only. This input is connected from the GStart pin of GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group.
Direction	BOOL	Direction	• Direction = 0: Motor runs in forward direction. • Direction = 0: Motor runs in reverse direction.
Stop	BOOL	Group stop	GStop = 1: **Stop** command to the motor, OP = 0 is given. Applicable in Auto mode only. This input is connected from the GStop pin of GrpCtrl or GrpStpMgt or RouteMgt or previous device in the group.

Parameter	Type	Description	Details
QStop	BOOL	Group quick stop	QStop = 1: **Stop** command to the motor, OP = 0 is given. Applicable in Auto mode only. This input is connected from the GQStop pin of GrpCtrl.
LStart1	BOOL	Local start in forward direction	LStart1 = 1: **Start** command in forward direction to the motor, OP = 1 is given. Applicable in Local mode only. This input is connected from the **Start** push button of local IO panel.
LStart2	BOOL	Local start in backward direction	LStart2 = 0...1: **Start** command in backward direction to the motor, OP = 1 is given. Applicable in Local mode only. This input is connected from the **Start** push button of local IO panel.
LStop	BOOL	Local stop	LStop = 0: **Stop** command to the motor, OP = 0 is given. Applicable in Local mode only. This input is connected from the **Stop** push button of local IO panel.
StartILck	BOOL	Starting interlock	• StartILck = 1: All the starting interlock conditions are operational. • StartILck = 0: All the starting interlock conditions are not operational. Applicable only in Auto and Manual modes of operation. Applicable only before starting of the motor. After motor is started, StartILck is not considered.
SftyILck	BOOL	Safety interlock	• SftyILck = 1: All the safety interlock conditions are operational. • SftyILck = 0: All the safety interlock conditions are not operational, trips the motor and generates an alarm. Applicable in all the modes of operation (Auto, Manual, and Local).

Parameter	Type	Description	Details
SeqILck	BOOL	Sequential interlock	• SeqILck = 1: All the sequential interlock conditions are operational. • SeqILck = 0: All the sequential interlock conditions are not operational, trips the motor and generates an alarm. Applicable in Auto mode only.
ProcILck	BOOL	Process interlock	• ProcILck = 1: All the process interlock conditions are operational. • ProcILck = 0: All the process interlock conditions are not operational, trips the motor and generates an alarm. Applicable in Auto and Manual modes only.
CondStart	BOOL	Conditional start	• CondStart = 1: **Start** command to the motor, OP = 1. • CondStart = 0: **Stop** command to the motor, OP = 0. Applicable in Auto mode only. If GActive = 1, CondStart is used to start or stop a motor based on process condition.
Avail	BOOL	Availability of motor	• Avail = 1: Motor is electrically available. • Avail = 0: Motor is electrically not available. **Stop** command to the motor, OP = 0. This signal comes from the MCC outgoing circuit (electrical protection active, circuit breaker switched off, drawer out, and so on).
OverLoad	BOOL	Overload	• OverLoad = 1: Motor is not in overload condition. • OverLoad = 0: Motor is in overload condition. **Stop** command to the motor, OP = 0. This signal comes from the MCC outgoing circuit (electrical protection active).

Parameter	Type	Description	Details
EStop	BOOL	Emergency stop	• EStop = 1: Plant process condition is operational for the motor to run. • EStop = 0: Emergency situation. Plant process condition is not operational for the motor to run. **Stop** command to the motor, OP = 0. Applicable in all operation modes (Auto, Manual, and Local). This input is connected from the **Emergency Stop** push button in the field.
FbFwd	BOOL	Forward Run feedback of motor - Electrical	Once the motor is given a **Start** command from output OP1, its electrical run feedback is continuously monitored. This signal comes from the MCC outgoing circuit. If OP = 1 and FbFwd = 0, then **Stop** command to the motor, OP = 0 is given and an alarm is generated after time delay. If OP = 0 and FbFwd = 1, an alarm is generated after time delay.
FbRev	BOOL	Reverse Run feedback of motor - Electrical	Once the motor is given a **Start** command from output OP2, its electrical run feedback is continuously monitored. This signal comes from the MCC outgoing circuit. If OP = 1 and FbRev = 0, then **Stop** command to the motor, OP = 0 is given and an alarm is generated after time delay. If OP = 0 and FbRev = 1, an alarm is generated after time delay.

Parameter	Type	Description	Details
FbProc	BOOL	Run feedback of motor - Process	Once the motor is given a **Start** command from output OP, its electrical run feedback is continuously monitored. This signal comes from Speed monitoring devices, Zero Speed switch, Proximity switches, and so on. If OP = 1 and FbProc = 0, then **Stop** command to the motor, OP = 0 is given, and an alarm is generated after time delay. If OP = 0 and FbProc = 1, an alarm is generated after time delay.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

The AlmInh parameter to inhibit alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the Mot2d block:

Parameter	Type	Description	Details
AlmInh	BOOL	Alarm inhibition	• AlmInh = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • AlmInh = 0: Alarms are not inhibited.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

To reset, alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the Mot2d block:

Parameter	Type	Description	Details
AlmRst	BOOL	Reset	• AlmRst = 1: Resets all the alarms generated by the block. • AlmRst = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from GRst pin of GrpCtrl block and resets the trip alarm.

The following table describes the output parameters of the Mot2d block:

Parameter	Type	Description	Details
Auto	BOOL	Auto mode	Auto = 1: Auto mode is selected. Motor operation is group based on GStart and GStop.
Manual	BOOL	Manual mode	Manual = 1: Manual mode is selected. Motor operation is based on commands **Start** and **Stop** given by operator from HMI.
Local	BOOL	Local mode	Local = 1: Local mode is selected. Motor operation is based on commands **LStart** and **LStop** given from field.
Oos	BOOL	Out of Service mode	Oos = 1: Out of Service mode is selected. Motor is not operated and OP = 0.
GActive	BOOL	Group active	The motor is started in the group and is active in the group. Start command pulse **Start** is obtained, GActive = 1.
OP1	BOOL	Output command to motor.	• OP1 = 0: Motor is given command to stop. • OP1 = 1: Motor is given command to start. The output switches on the contactor or circuit breaker controlling the motor in forward direction.

Parameter	Type	Description	Details
OP2	BOOL	Output command to motor.	• OP2 = 0: Motor is given command to stop. • OP2 = 1: Motor is given command to start. The output switches on the contactor or circuit breaker controlling the motor in forward direction.
SeqStart	BOOL	Start command to next motor in group.	Start command (pulse) is received and outputs OP and Run = 1, sequential start command pulse is given to the next motor connected in the group.
SeqStop	BOOL	Stop command to next motor in group.	Stop or QStop = 1 and OP and Run = 0, sequential stop command is given to the next motor connected in the group.
ILck	BOOL	Interlock status	• ILck = 0: All the interlocks are not operational. There is an active interlock present. • ILck = 1: All the interlocks are operational.
Ready	BOOL	Motor ready to start.	• Ready = 0: The conditions to start the motor are not operational. Motor will not be given **Start** command. • Ready = 1: All the conditions to start the motor are operational.
Run	BOOL	Motor running	Motor is running. This output is given if the motor is running and run feedback (FbFwd) or (FbRev) is received. Applicable in all the operation modes.
FwdRun	BOOL	Forward run	Motor is running in forward direction.
RevRun	BOOL	Reverse run	Motor is running in reverse direction.

Parameter	Type	Description	Details
Warn	BOOL	Alert	• Warn = 1: The motor is operational with no active alert present. • Warn = 0: The motor is operational with an active alert present.
Trip	BOOL	Trip	• Trip = 1: The motor is operational with no active trip alarm present. • Trip = 0: The motor has tripped with an active trip alarm present.

The following table describes the input/output parameter of the Mot2d block:

Parameter	Type	Description
Stx	Mot2d_St	Status and control of the motor in 2 directions. Refer to the Structure to interface with the HMI *(see page 158)*.

Mot2d - Representation as SCoD in UAG

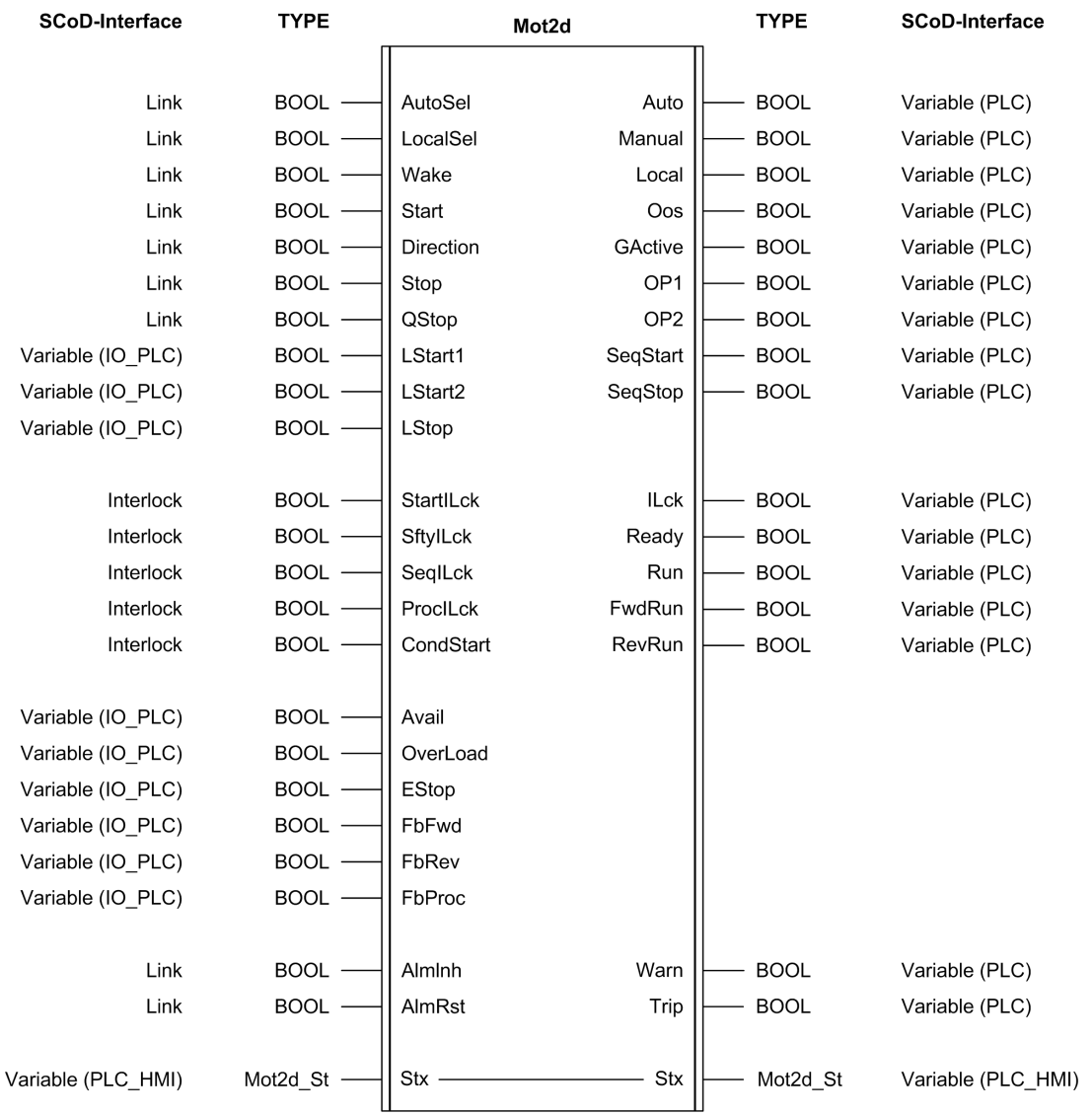
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the Stx variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type HMI is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the Mot2d block.


SCoD-Interface
TYPE
Mot2d
TYPE
SCoD-Interface
Link BOOL AutoSel Auto BOOL Variable (PLC)
Link BOOL LocalSel Manual BOOL Variable (PLC)
Link BOOL Wake Local BOOL Variable (PLC)
Link BOOL Start Oos BOOL Variable (PLC)
Link BOOL Direction GActive BOOL Variable (PLC)
Link BOOL Stop OP1 BOOL Variable (PLC)
Link BOOL QStop OP2 BOOL Variable (PLC)
Variable (IO_PLC) BOOL LStart1 SeqStart BOOL Variable (PLC)
Variable (IO_PLC) BOOL LStart2 SeqStop BOOL Variable (PLC)
Variable (IO_PLC) BOOL LStop
Interlock BOOL StartILck ILck BOOL Variable (PLC)
Interlock BOOL SftyILck Ready BOOL Variable (PLC)
Interlock BOOL SeqILck Run BOOL Variable (PLC)
Interlock BOOL ProcILck FwdRun BOOL Variable (PLC)
Interlock BOOL CondStart RevRun BOOL Variable (PLC)
Variable (IO_PLC) BOOL Avail
Variable (IO_PLC) BOOL OverLoad
Variable (IO_PLC) BOOL EStop
Variable (IO_PLC) BOOL FbFwd
Variable (IO_PLC) BOOL FbRev
Variable (IO_PLC) BOOL FbProc
Link BOOL AlmInh Warn BOOL Variable (PLC)
Link BOOL AlmRst Trip BOOL Variable (PLC)
Variable (PLC_HMI) Mot2d_St Stx Stx Mot2d_St Variable (PLC_HMI)

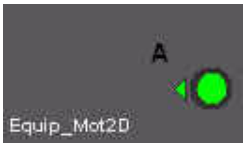
SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Run feedback delay (sec)	`Stx.TRunFb`	0	**Configuration**	2	Sets run feedback delay (s).
Process feedback delay (sec)	`Stx.TProcFb`	0	**Configuration**	3	Sets process feedback delay (s).
Loss of process fb delay (sec)	`Stx.TLProcFb`	0	**Configuration**	3	Sets loss of process feedback delay (s).
Inv direction start delay (sec)	`Stx.TInv`	0	**Configuration**	9	Sets inverse direction start delay (s).
Display format (time)	`Stx.RunHr`	###.#	**Configuration**	1	Sets the format for display in HMI genie and popup.
Area Information	`AraeInfo`	Not available	**Device Data**	1	Area information
Asset ID	`AssetId`	Not available	**Device Data**	7	Asset ID

Mot2D - Representation in the HMI

Motor 2 Direction (Genie)

The Mot2D block is represented in the HMI by the following genie.



Genie Elements

Symbol	Meaning
	Opens the Mot2D popup.
	The instance name of the Mot2D block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

Animation Description of the Genie

Symbol	Animation	Meaning
	Line color: Steady light gray Fill color: Steady light gray	Motor is in Sleep mode.
	Line color: Steady dark gray Fill color: Steady white	• Motor is awake or in Local mode or Manual mode and ready. • Motor is awake or in Local mode or Manual mode and ready after alarms are reset.
	Line color: Dark gray Fill color: Light gray	Motor is awake or in Local mode or Manual mode and not ready.
	Line color: Steady dark gray Fill color: Flashing green	Motor is started and feedback is yet to be obtained or motor is stopping.
	Line color: Steady dark gray Fill color: Steady green	Motor is running.
	Line color: Flashing yellow Fill color: Steady green	Motor is running and an alert alarm is active.

Symbol	Animation	Meaning
	Line color: Steady yellow Fill color: Steady green	Motor is running and an alert alarm is acknowledged.
	Line color: Steady dark gray Fill color: Flashing red	Motor is tripped and a trip alarm is active.
	Line color: Steady dark gray Fill color: Steady red	Motor is tripped and a trip alarm is acknowledged.

Modes

On the top left corner of the motor 2 direction genie, the mode of operation is indicated.

The following table describes the mode of operations:

Mode	Meaning
A	Motor is operating in the Auto mode.
M	Motor is operating in the Manual mode.
L	Motor is operating in the Local mode.
O	Motor is in Out of Service mode.

Item	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Direction

Symbol	Meaning
	Motor is in forward direction.
	Motor is in reverse direction.

Alarm Masking

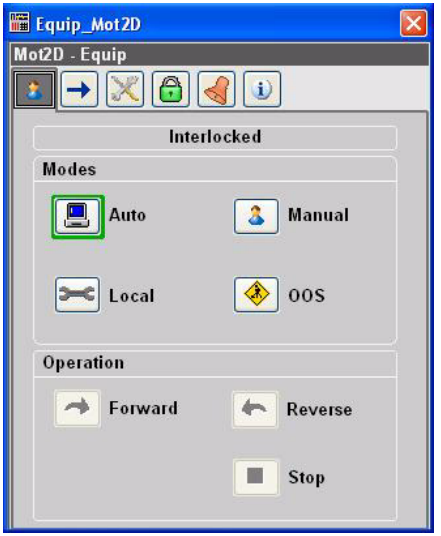
Symbol	Meaning
	Trip or alert alarms in the motor 2 direction are masked.

Popup

The `Mot2D` block popup has the following tabs:

- **Operator** tab
- **Extended** tab
- **Configuration** tab
- **Interlock** tab
- **Alarm** tab
- **Information** tab

Operator Tab



This tab has 3 sub-sections:

- **Status**
- **Modes**
- **Operation**

The sub-section **Status** displays the current status.

The possible displays are listed as follows:

- **Sleeping**
- **Group Active**
- **Ready**
- **Not Ready**
- **Starting Forward**
- **Starting Reverse**
- **Stopping**
- **Running Forward**
- **Running Reverse**

- **Interlocked**
- **Warning**
- **Trip**

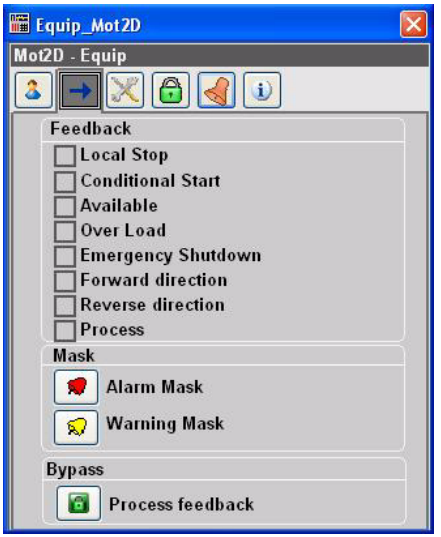
The following table describes the sub-section **Modes**:

Item	Description
Auto	Switches to Auto operation mode controlled by the PLC program.
Manual	Switches to Manual operation mode controlled by the HMI.
Local	Switches to Local operation mode controlled by the local control box of the motor.
OOS	Switches to Out of Service mode for maintenance purpose of the object (applicable in all modes).

The following table describes the sub-section **Operation**:

Item	Description
Forward	Motor runs in forward direction in Manual mode only.
Reverse	Motor runs in reverse direction in Manual mode only.
Stop	Stops the operation of the device in Manual mode only.

Extended Tab



This tab has 2 sub-sections:

- **Feedback**
- **Mask**
- **Bypass**

The following table describes the sub-section **Feedback**:

Item	Animation	Description
Local Stop	Steady green	Motor locally stopped feedback is obtained.
	Steady gray	Motor locally not stopped feedback is obtained.
Conditional Start	Steady green	Motor is started.
	Steady gray	Motor is not started.
Available	Steady green	Motor electrically available feedback is obtained.
	Steady gray	Motor electrically not available feedback is obtained.
Over Load	Steady green	Motor overload feedback is obtained.
	Steady gray	Motor overload feedback is not obtained.
Emergency Shutdown	Steady green	Emergency stop feedback is obtained.
	Steady gray	Emergency stop feedback is not obtained.
Forward direction	Steady green	Motor is active and running in forward direction.
	Steady gray	Motor is not active.
Reverse direction	Steady green	Motor is active and running in reverse direction.
	Steady gray	Motor is not active.
Process	Steady green	Process feedback is obtained.
	Steady gray	Process feedback is not obtained.

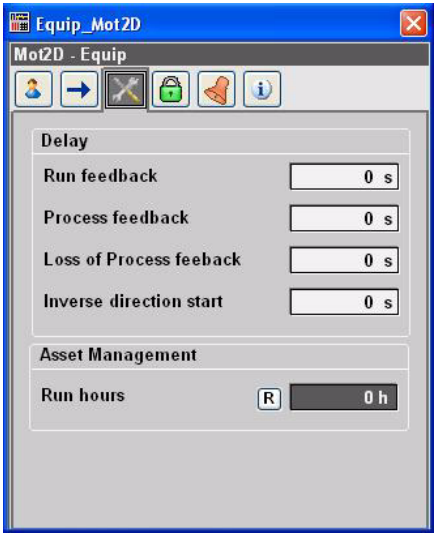
The following table describes the sub-section **Mask**:

Item	Description
Alarm Mask	Masks the indicated current trip alarms of this block in the alarm list.
Warning Mask	Masks the current alerts of this block in the alarm list.

The following table describes the sub-section **Bypass**:

Item	Description
Process feedback	Bypasses the process feedback.

Configuration Tab



This tab has 2 sub-sections:

- **Delay**
- **Asset Management**

The following table describes the sub-section **Delay**:

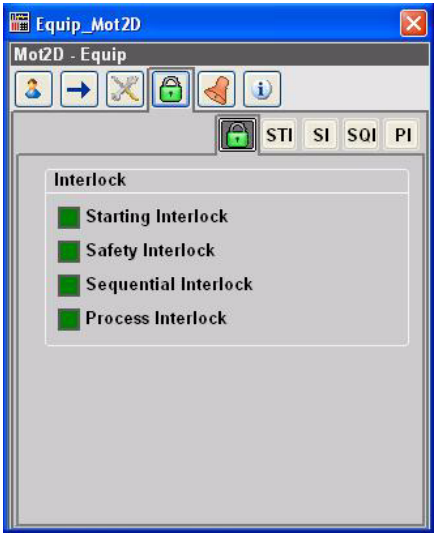
Item	Description
Run feedback	Displays run feedback delay of motor, configurable from HMI. After starting motor, if running feedback is not received before delay time, an alarm is generated, and the motor is tripped.
Process feedback	Displays process feedback delay of motor, configurable from HMI. After starting motor, if process feedback is not received before delay time, an alarm is generated, and the motor is tripped.

Item	Description
Loss of Process feedback	Displays loss of process feedback delay of motor, configurable from HMI. After receiving process feedback, if there is loss of feedback while motor running and is not received before delay time, an alarm is generated, and the motor is tripped.
Inverse direction start	Inverse direction starts delay of motor, configurable from HMI. When the time taken to switch on motor from forward to reverse direction or vice versa exceeds delay time, an alarm is generated, and the motor is tripped.

The following table describes the sub-section **Asset Management**:

Item	Description
Run hours	Displays the run hours of the motor.
	Resets the run hours value of the motor to 0.

Interlock Tab



This tab has a sub-section called as **Interlock**, which is described in the following table:

Item	Animation	Description
Starting Interlock	Steady green	**Starting Interlock** is active, and the related device is not blocked.
	Steady gray	**Starting Interlock** is not active, and the related device is blocked.
Safety Interlock	Steady green	**Safety Interlock** is active, and the related device is not blocked.
	Steady gray	**Safety Interlock** is not active, and the related device is blocked.
Sequential Interlock	Steady green	**Sequential Interlock** is active, and the related device is not blocked.
	Steady gray	**Sequential Interlock** is not active, and the related device is blocked.
Process Interlock	Steady green	**Process Interlock** is active, and the related device is not blocked.
	Steady gray	**Process Interlock** is not active, and the related device is blocked.

This tab has a sub-tabs Interlock, **STI**, **SI**, **SQI**, and **PI**.

For description about **Interlock** sub-tabs, refer to Commonly Used Tabs *(see page 53)*.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

Mot2d_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type Mot2d_St for the Mot2d block:

Element	Type	Description
RunHr	INT	Run hours of motor
TRunFb	INT	Run feedback delay
TProcFb	INT	Process feedback delay
TLProcFb	INT	Loss of process feedback delay
TInv	INT	Inverse direction start delay
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Stw2	WORD	Status word 2 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI
Dummy	WORD	Not used

Bit description for the Ctw element:

Bit	Description
0	Auto mode selection from HMI
1	Manual mode selection from HMI
2	Local mode selection from HMI
3	Out of Service mode selection from HMI
4	Start command from HMI
5	Start2 command from HMI
6	Stop command from HMI
7	Alarm reset from HMI
8	Run hour reset from HMI
9	Bypass process feedback from HMI
10...13	Not used
14	Alert mask
15	Alarm mask

Bit description for the Stw1 element:

Bit	Description
0	Auto mode selection from HMI
1	Local mode selection from HMI
2	Group wake
3	Local stop
4	Starting interlock
5	Safety interlock
6	Sequential interlock
7	Process interlock
8	Motor electrically available
9	Motor overload
10	Emergency stop
11	Run feedback forward direction
12	Run feedback reverse direction
13	Process feedback
14	Alarm inhibition
15	Not used

Bit description for the Stw2 element:

Bit	Description
0	Auto mode active status
1	Manual mode active status
2	Local mode active status
3	Out of Service mode active status
4	Output OP1 active
5	Output OP2 active
6	Motor ready status
7	Motor running status
8	Forward run
9	Reverse run
10	Conditional start active status
11	Not used
12	Interlock active status
13	Group active status

Bit	Description
14	Alert active status
15	Trip active status

Bit description for the `Alw1` element:

Bit	Description
0	Motor locally stop
1	Safety interlock not available
2	Sequential interlock not available
3	Process interlock not available
4	Motor electrically not available
5	Motor overload
6	Motor emergency stop
7	Run feedback not active when motor is running
8	Run feedback active when motor is not running
9	Reverse feedback low when motor is running
10	Reverse feedback high when motor is not running
11	Process feedback not active when motor is running
12	Process feedback active when motor is not running
13	Not used
14	Run hours maximum value reached
15	Active trip alarm

8.3 AnlgIn: Analog Input

Overview

This section describes the AnlgIn block.

What's in this Section?

This section contains the following topics:

Topic	Page
AnlgIn - Description	162
AnlgIn - Representation as SCoD in UAG	167
AnlgIn - Representation in the HMI	169
AnlgIn_St Derived Data Type	172

AnlgIn - Description

General

The AnlgIn block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB AnlgIn
Vijeo Citect	Genie AnlgIn
UAG	SCoD AnlgIn

Function Description

The AnlgIn block is used to read analog inputs from the analog input modules.

This block can be used for any type of input (4...20 mA as well as PT100 inputs). It can also be used to trigger alarms from calculated signals which are not coming from input modules.

A wake-up command is given by the GrpCtrl block to wake-up the analog input block from Sleep mode. The analog input signal comes from the analog input module and this value is a raw data. The high and low range of this raw value can be configured. The output of this block is the analog processed value.

The high and low range of analog processed value can be configured. The cut off ranges, that is, high cut off and low cut off values resulting from the scaling calculation can be configured. High cut off and above which the value entered into the HPV input must be considered as the measurement value (APV output). Low cut off, and below which the value entered into the LPV input must be considered as the measurement value (APV output).

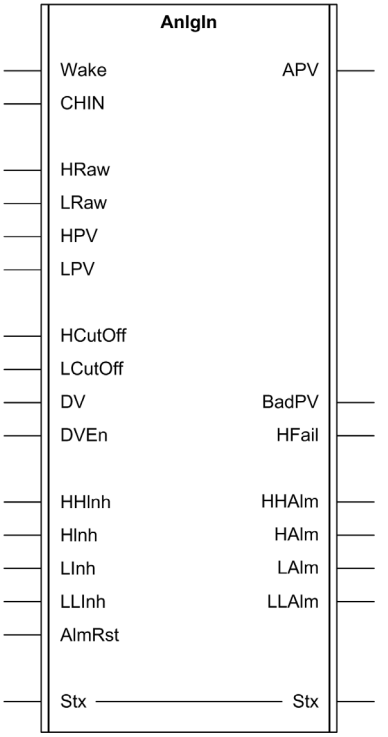
Very high (**High High SP**), high (**High SP**), low (**Low SP**), very low (**Low Low SP**) limits for alarms and hysteresis for alarm limits check can be configured. A default value can be configured which would be considered when the input channel is inoperable if the default value is enabled.

When the simulation is switched on, the simulation value is set from the popup (**Set Point**) which is considered as the input and moved to APV.

The alert and trip is indicated in the SCADA. The number of active alarms is shown in the popup of the AnlgIn block. The details of the active alarms are shown in the **Alarm** tab of the popup. Based on the configured conditions for very high, high, low, and very low inhibition pins, the alarms for this block can be inhibited. The alert and trip alarms can be reset. Bad processed value, potential hardware alarms and indications are obtained.

Representation in FBD

The following figure shows the FBD representation of the `AnlgIn` block.



Parameter Description

The following table describes the input parameters of the `AnlgIn` block:

Parameter	Type	Description	Details
`Wake`	`BOOL`	Group wake-up	Group wake-up and request for group availability check: • `Wake` = 0: Group is in Sleep mode. • `Wake` = 1: Group is active. This input is connected from output `GWake` of `GrpCtrl`.
`CHIN`	`INT`	Channel input	Analog input signal Normally corresponds to a value in raw data coming from the input/output peripherals.
`HRaw`	`INT`	High raw value	High range of channel input signal.

Parameter	Type	Description	Details
LRaw	INT	Low raw value	Low range of channel input signal.
HPV	REAL	High process value	High range of the measurement in engineering units (APV output) corresponding to the maximum value (configured in the HRaw input) of the CHIN input.
LPV	REAL	Low process value	Low range of the measurement in engineering units (APV output) corresponding to the minimum value (configured in the LRaw input) of the CHIN input.
HCutOff	REAL	High cut off	Value (in engineering units) resulting from the scaling calculation, and above which the value entered into the HPV input must be considered as the measurement value (APV output).
LCutOff	REAL	Low cut off	Value (in engineering units) resulting from the scaling calculation, and below which the value entered into the LPV input must be considered as the measurement value (APV output).
DV	REAL	Default value	Value that must be considered in engineering units (APV output signal) when the channel is inoperable.
DVEn	BOOL	Default value enable	Enabling or disabling of the corresponding input signal.
HHInh	BOOL	Inhibition of high high alarm	Inhibition of high high alarm
HInh	BOOL	Inhibition of high alarm	Inhibition of high alarm
LInh	BOOL	Inhibition of low alarm	Inhibition of low alarm
LLInh	BOOL	Inhibition of low low alarm	Inhibition of low low alarm

⚠ CAUTION
UNINTENDED MACHINE OR PROCESS BEHAVIOR Be sure of the state of security of your machine or process environment. To reset, alarms must be used very carefully. **Failure to follow these instructions can result in injury or equipment damage.**

The following table describes the input parameter of the `AlgIn` block:

Parameter	Type	Description	Details
`AlmRst`	`BOOL`	Reset	• `AlmRst` = 1: Resets all the alarms generated by the block. • `AlmRst` = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from `GRst` pin of `GrpCtrl` block and resets the trip alarm.

The following table describes the output parameters of the `AnlgIn` block:

Parameter	Type	Description	Details
`APV`	`REAL`	Analog processed value	Analog processed value.
`BadPV`	`BOOL`	Bad process value	`BadPV` = 1: The input signal value is not good.
`HFail`	`BOOL`	Channel hardware inoperable	If `DVEn` = 1, then `HFail` = 1.
`HHAlm`	`BOOL`	High high level alarm	High high level alarm. Alarm is generated only if `Wake` is high.
`HAlm`	`BOOL`	High level alarm	High level alarm. Alarm is generated only if `Wake` is high.
`LAlm`	`BOOL`	Low level alarm	Low level alarm. Alarm is generated only if `Wake` is high.
`LLAlm`	`BOOL`	Low low level alarm	Low low level alarm. Alarm is generated only if `Wake` is high.

The following table describes the input/output parameter of the AnlgIn block:

Parameter	Type	Description
Stx	AnlgIn_St	Status and control of the AnlgIn block. Refer to the Structure to interface with the HMI *(see page 172)*.

AnlgIn - Representation as SCoD in UAG

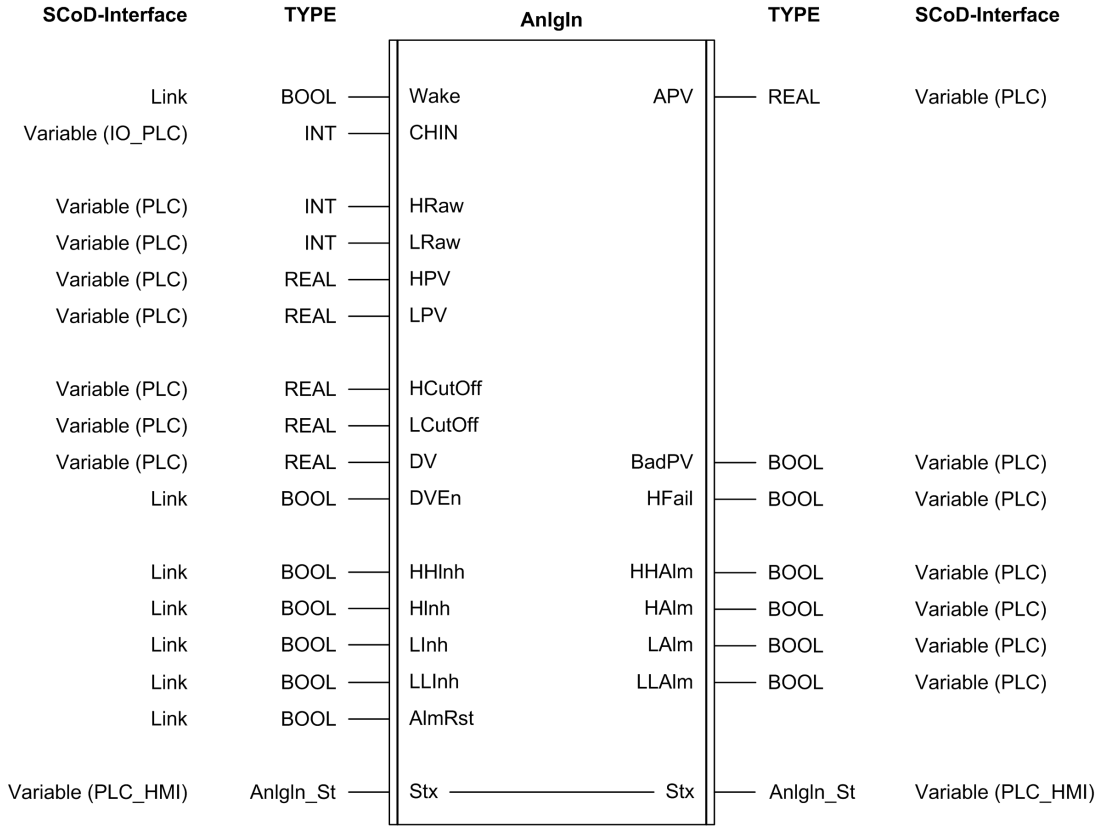
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the Stx variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type HMI is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the AnlgIn block.



SCoD Properties

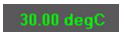
Name	Variable	Initial Value	Property Tab	Position	Description
Display format	Stx.APV	###.##	**Configuration**	1	Sets the format for display in HMI genie and popup.
Measurement unit	Stx.APV	%	**Configuration**	7	Sets measurement unit.
Default value	DV	0.0	**Configuration**	2	Sets the default value.
Lag filter delay (sec)	Stx.Lag	0.0	**Configuration**	8	Lag filter delay
High raw value	HRaw	100	**Limits**	7	Sets high raw value.
Low raw value	LRaw	0	**Limits**	1	Sets low raw value.
High process value	HPV	100.0	**Limits**	8	Sets high PV value.
Low process value	LPV	0.0	**Limits**	2	Sets low PV value.
High cutoff value	HCutOff	100.0	**Limits**	9	Sets high cut off.
Low cutoff value	LCutOff	0.0	**Limits**	3	Sets low cut off.
High high limit value	Stx.HHSP	Not available	**Limits**	11	Sets high high limit for alarm.
High limit value	Stx.HSP	Not available	**Limits**	10	Sets high limit for alarm.
Low limit value	Stx.LSP	Not available	**Limits**	4	Sets low limit for alarm.
Low low limit value	Stx.LLSP	Not available	**Limits**	5	Sets low low limit for alarm.
Hystersis	Stx.Hyst	Not available	**Limits**	6	Sets hysteresis value.
Eng.Max	Stx.APV	0.0	**Range**	7	Sets the maximum Eng. value.
Eng.Min	Stx.APV	100.0	**Range**	1	Sets the minimum Eng. value.
Area Information	AreaInfo	Not available	**Device Data**	1	Area information.
Asset ID	AssetId	Not available	**Device Data**	7	Asset ID.

AnlgIn - Representation in the HMI

Analog Input (Genie)

The AnlgIn block is represented in the HMI by the following genie.

Genie Elements

Symbol	Meaning
	Displays the analog input processed value and opens the AnlgIn popup.
	The instance name of the AnlgIn block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

Animation Description of the Genie

Symbol	Animation	Meaning
	Line color: Flashing purple	Hardware inoperable in the anlaog input block is active.
	Text color: Steady purple	Hardware inoperable in the anlaog input block is acknowledged.
	Text color: Flashing yellow	High or low alert is active.
	Text color: Steady yellow	High or low alert is acknowledged.
	Text color: Flashing red	High high or low low alarm is active.
	Text color: Steady red	High high or low low alarm is acknowledged.

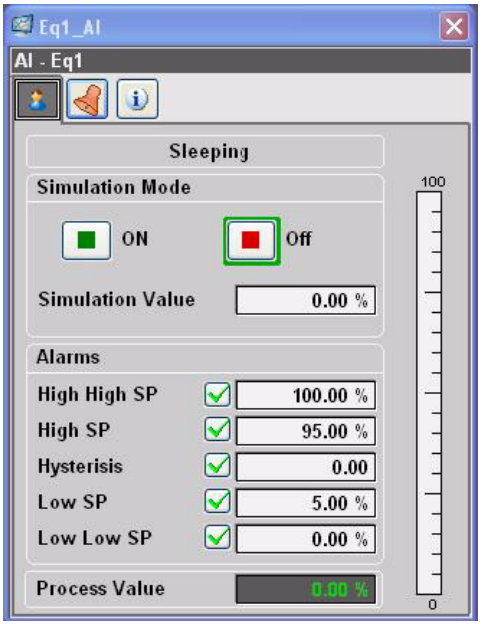
Item	Meaning
S	S is displayed on the left top corner when the simulator is active.
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Popup

The `AnlgIn` block popup has the following tabs:

- **Operator** tab
- **Alarm** tab
- **Information** tab

Operator Tab



This tab has 3 sub-sections:

- **Status**
- **Simulation Mode**
- **Alarms**

The sub-section **Status** displays the current status.

The possible displays are listed as follows:

- **Sleeping**
- **Group Active**
- **Hardware Failure**
- **Bad Process Value**
- **Alarm**
- **Warning**

The following table describes the sub-section **Simulation Mode**:

Item	Description
ON	Switch on the simulation.
Off	Switch off the simulation.
Simulation Value	Enter the simulation value or set point.

The following table describes the sub-section **Alarms**:

Item	Description
High High SP	Enter the very high set point. Select the check box to enable the very high set point alarm. Click to clear the check box to disable the very high set point alarm.
High SP	Enter the high set point. Select the check box to enable the high set point alarm. Click to clear the check box to disable the high set point alarm.
Hysteresis	Enter the hysteresis value. Select the check box to enable the hysteresis. Click to clear the check box to disable the hysteresis.
Low SP	Enter the low set point. Select the check box to enable the low set point alarm. Click to clear the check box to disable the low set point alarm.
Low Low SP	Enter the very low set point. Select the check box to enable the very low set point alarm. Click to clear the check box to disable the very low set point alarm.

Item	Description
Process Value	Displays the process value with bar graph.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

AnlgIn_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type AnlgIn_St for the AnlgIn block:

Element	Type	Description
SimV	REAL	Simulation value
APV	REAL	Analog process value
SPHH	REAL	High high limit for an alarm
SPH	REAL	High limit for an alarm
SPL	REAL	Low limit for an alarm
SPLL	REAL	Low low limit for an alarm
Hyst	REAL	Hysteresis
TON	DINT	Timer on
TOFF	DINT	Timer off
Lag	INT	Lag filter time
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Alw1	WORD	Alarm word 1to interact with HMI

Bit description for the Ctw element:

Bit	Description
0	Simulation selection
1	Not used
2	High high alarm enable
3	High alarm enable
4	Low alarm enable
5	Low low alarm enable
6	Hysteresis enable
7	Reset
8...15	Not used

Bit description for the `Stw1` element:

Bit	Description
0	Wake-up
1...15	Not used

Bit description for the `Alw1` element:

Bit	Description
0	Hardware inoperable
1...2	Not used
3	Bad process value
4	High high alarm
5	High alarm
6	Low alarm
7	Low low alarm
8...15	Not used

8.4 AnlgOut: Analog Output

Overview

This section describes the AnlgOut block.

What's in this Section?

This section contains the following topics:

Topic	Page
AnlgOut - Description	175
AnlgOut - Representation as SCoD in UAG	179
AnlgOut - Representation in the HMI	181
AnlgOut_St Derived Data Type	184

AnlgOut - Description

General

The AnlgOut block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB AnlgOut
Vijeo Citect	Genie AnlgOut
UAG	SCoD AnlgOut

Function Description

This AnlgOut block reads analog inputs from the set point coming from a PIDB (Proportional, Integral and Derivative Controller Block) or a calculation block, which sends the analog processed value to analog output modules.

When the AnlgOut block is in Auto mode, a wake-up **wake** command is given by the GrpCtrl block to wake-up the analog output block from Sleep mode. The analog set point comes from the calculation or PIDB block. The set point value is scaled to high and low range of raw value output based on high and low set point, and an analog output is given to the respective channel of the analog output module. The output is restricted to high and low limit.

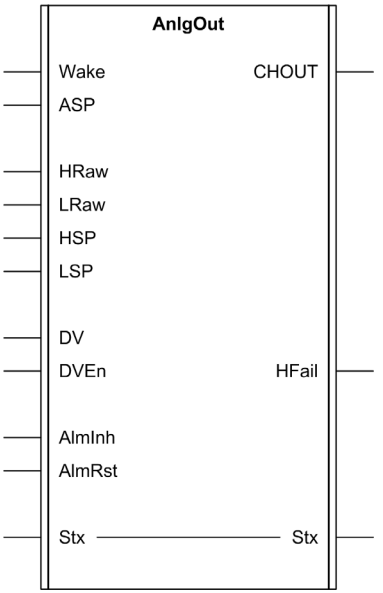
When the default value is enabled (DVEn), the default value is moved to the output. When the simulation is switched on, the simulation value is set from the popup (**Set Point**) which is considered as the input and moved to calculated value of the output CHOUT.

The alerts and trips are indicated in the SCADA. The number of active alarms is shown in the popup of the AnlgOut block. The details of the active alarms is shown in the **Alarm** tab of the popup. In addition, the hardware inoperable alarms and indications are obtained.

Based on the user configured conditions, the alarms for this block can be inhibited by connecting the logic to the input pin (AlmInh). The input pin (AlmRst) connected from GrpCtrl block or the operator from the popup resets the trip alarms.

Representation in FBD

The following figure shows the FBD representation of the `AnlgOut` block.



Parameter Description

The following table describes the input parameters of the `AnlgOut` block:

Parameter	Type	Description	Details
`Wake`	`BOOL`	Group wake-up	Group wake-up and request for group availability check: • `Wake` = 1: Group is active. • `Wake` = 0: Group is sleeping. This input is connected from the `GWake` output of `GrpCtrl`.
`ASP`	`REAL`	Analog set point	Analog set point.
`HRaw`	`INT`	High raw value	High range of channel input value.
`LRaw`	`INT`	Low raw value	Low range of channel input value.
`HSP`	`REAL`	High set point	High range of set point value to be scaled to raw value.
`LSP`	`REAL`	Low set point	Low range of set point value to be scaled to raw value.

Parameter	Type	Description	Details
DV	REAL	Default value	Value that must be considered in engineering units (CHOUT output signal), when the channel is inoperable.
DVEn	BOOL	Default value enable	Enabling or disabling of the corresponding input signal.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

The AlmInh parameter to inhibit alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the AnlgOut block:

Parameter	Type	Description	Details
AlmInh	BOOL	Alarm inhibition	• AlmInh = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • AlmInh = 0: Alarms are not inhibited.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

To reset, alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the AnlgOut block:

Parameter	Type	Description	Details
AlmRst	BOOL	Reset	• AlmRst = 1: Resets all the alarms generated by the block. • AlmRst = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from GRst pin of GrpCtrl block and resets the trip alarm.

The following table describes the output parameters of the AnlgOut block:

Parameter	Type	Description	Details
CHOUT	REAL	Channel output	This gives the analog processed value to the channel of analog output module.
HFail	BOOL	Channel hardware inoperable	If DVEn = 1, then HFail = 1.

The following table describes the input/output parameter of the AnlgOut block:

Parameter	Type	Description
Stx	AnlgOut_St	Status and control of the AnlgOut block. Refer to the Structure to interface with the HMI *(see page 184)*.

AnlgOut - Representation as SCoD in UAG

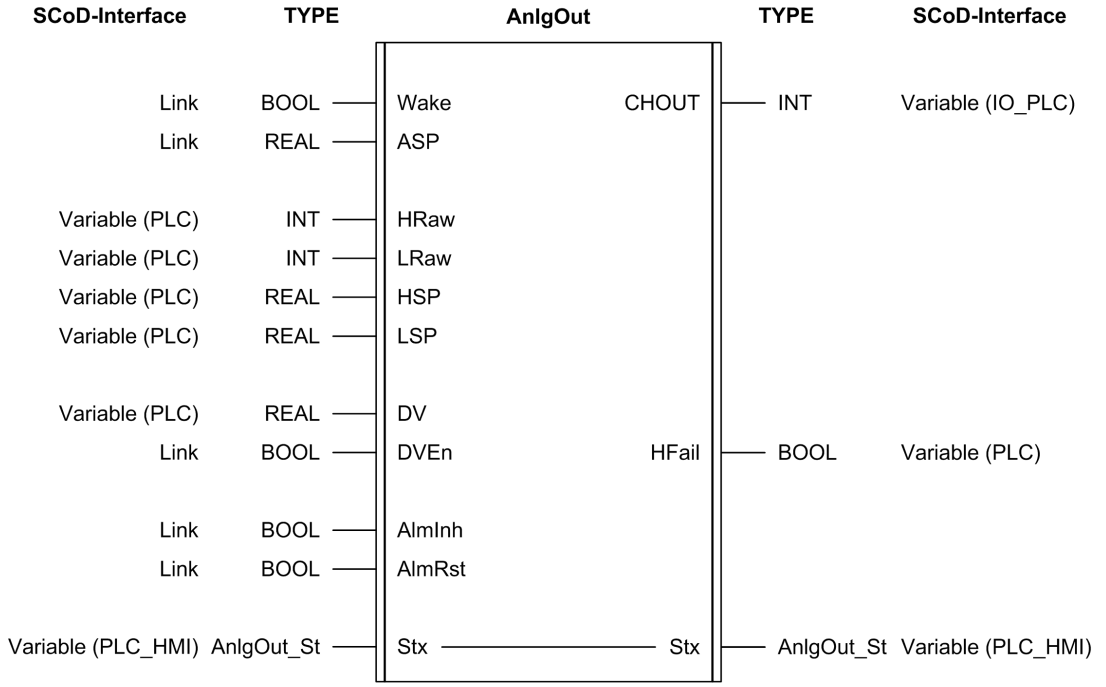
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the Stx variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type HMI is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the AnlgOut block.



SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Default value	DV	0.0	**Configuration**	2	Default value enable
Display format	Stx.SimV	###.##	**Configuration**	1	Display format
Measurement Unit	Stx.SimV	%	**Configuration**	7	Measurement value
High Limit	Stx.SPH	100.0	**Limits**	9	High limit
High set point	HSP	100.0	**Limits**	8	High set point
Low Limit	Stx.SPL	0.0	**Limits**	3	Low limit
Low set point	LSP	0.0	**Limits**	2	Low set point
Raw value maximum	HRaw	100.0	**Limits**	7	High range of channel input value
Raw value minimum	LRaw	0.0	**Limits**	1	Low range of channel input value
Area Information	AreaInfo	Not available	**Device Data**	1	Area information
Asset ID	AssetId	Not available	**Device Data**	7	Asset ID

AnlgOut - Representation in the HMI

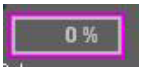
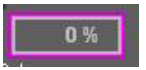
Analog Output (Genie)

The AnlgOut block is represented in the HMI by the following genie:

Genie Elements

Symbol	Meaning
	Displays the AnlgOut output value and opens the AnlgOut output popup.
	The instance name of the AnlgOut block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

Animation Description of the Genie

Symbol	Animation	Meaning
	Line color: Flashing purple	Hardware inoperable in the AnlgOut block is active.
	Line color: Steady purple	Hardware inoperable in the AnlgOut block is acknowledged.

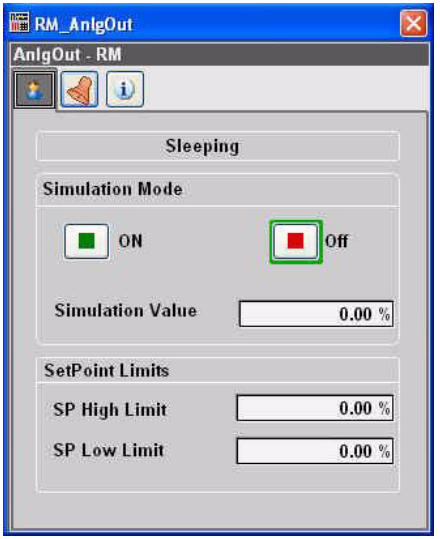
Symbol	Meaning
S	**S** is displayed on the top left corner when the simulator is in Active mode.
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Popup

The AnlgOut block popup has the following tabs:

- **Operator** tab
- **Alarm** tab
- **Information** tab

Operator Tab



This tab has 2 sub-sections:

- **Simulation Mode**
- **SetPoint Limits**

The sub-section **Status** displays the current status.

The possible displays are listed as follows:

- **Sleeping**
- **Group Active**
- **Hardware Failure**

The following table describes the sub-section **Simulation Mode**:

Item	Description
ON	Switch on the simulation.
Off	Switch off the simulation.
Simulation Value	Enter the simulation value or set point through set point entry.

The following table describes the sub-section **SetPoint Limits**:

Item	Description
SP High Limit	Displays set point high limit.
SP Low Limit	Displays set point low limit.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

AnlgOut_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type AnlgOut_St for the AnlgOut block:

Element	Type	Description
SimV	REAL	Simulation value
SPH	REAL	Set point high limit
SPL	REAL	Set point low limit
CHOUT	INT	Channel output
Ctw	WORD	Control word to interact with HMI
stw1	WORD	Status word 1 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI

Bit description for the Ctw element:

Bit	Description
0	Simulation selection
1...6	Not used
7	Alarm reset
8...15	Not used

Bit description for the stw1 element:

Bit	Description
0	Wake-up
1...15	Not used

Bit description for the Alw1 element:

Bit	Description
0	Hardware inoperable
1...15	Not used

8.5 DigIn: Digital Input

Overview

This section describes the DigIn block.

What's in this Section?

This section contains the following topics:

Topic	Page
DigIn - Description	186
DigIn - Representation as SCoD in UAG	190
DigIn - Representation in the HMI	192
DigIn_St Derived Data Type	196

DigIn - Description

General

The DigIn block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB DigIn
Vijeo Citect	Genie DigIn
UAG	SCoD DigIn

Function Description

The DigIn block is used to read digital inputs from the digital input modules and to monitor the input module.

This block is used in a group to manage the alarm circuits which are not directly managed by other blocks. These alarm circuits can be either of alert type or of trip type. It can also be used to trigger alarms from calculated signals which are not coming from input modules.

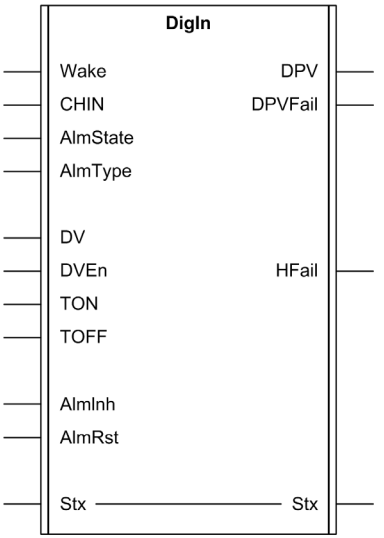
A wake-up command is given by the GrpCtrl block to wake-up the digital input block from Sleep mode. If it is not Wake, the alarm will not get generated. The digital input signal comes from the digital input module. The output of this block is the digital processed value.

You can configure the timing of digital input connection and disconnection for the channel input. When the hardware is inoperable, the default value is considered as the input if the default value is enabled. When the digital processed value is same as the alarm state and alarm is enabled, an alarm is generated. Alarm can be enabled/disabled from HMI popup. The type of alarm can be configured, that is, alert or trip. When the simulation is switched on, the simulation value is set from the popup is considered as the digital input and moved to DPV.

The alert and trip are indicated in the SCADA. The number of active alarms is shown in the popup of digital input block. The details of the active alarms are shown in the **Alarm** tab of the popup. Inoperable hardware alarms and indications are obtained. The alarms for digital input block can be inhibited by connecting the logic to the input pin (AlmInh). The input pin (AlmRst) connected from GrpCtrl block or the operator from the popup resets the trip alarm.

Representation in FBD

The following figure shows the FBD representation of the `DigIn` block.



Parameter Description

The following table describes the input parameters of the `DigIn` block:

Parameter	Type	Description	Details
`Wake`	`BOOL`	Group wake-up	Group wake-up and request for group availability check: ● `Wake` = 1: Group is active. ● `Wake` = 0: Group is in Sleep mode. This input is connected from output `Wake` of `GrpCtrl` or `RouteMgt`
`CHIN`	`BOOL`	Channel input	Digital input signal from the DI module.
`AlmState`	`BOOL`	Alarm state	An alarm is generated when `DPV` = `AlmState`.
`AlmType`	`BOOL`	Alarm type	Type of alarm: ● `AlmType` = 1: The signal is a trip alarm. ● `AlmType` = 0: The signal is an alert alarm.

Parameter	Type	Description	Details
DV	BOOL	Default value	Default value. It overwrites the input signal when a hardware becomes inoperable. This value is considered when a channel becomes inoperable.
DVEn	BOOL	Default value enabled	Enables the default value.
TON	DINT	Timer on	Timing set point for digital input connection (CHIN) to activate in ms.
TOFF	DINT	Timer off	Timing set point for digital input connection (CHIN) to deactivate in ms.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

The AlmInh parameter to inhibit alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the DigIn block:

Parameter	Type	Description	Details
AlmInh	BOOL	Alarm inhibition	• AlmInh = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • AlmInh = 0: Alarms are not inhibited.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

To reset, alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the DigIn block:

Parameter	Type	Description	Details
AlmRst	BOOL	Reset	• AlmRst = 1: Resets all the alarms generated by the block. • AlmRst = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from GRst pin of GrpCtrl block and resets the trip alarm.

The following table describes the output parameters of the DigIn block:

Parameter	Type	Description	Details
DPV	BOOL	Digital processed value	Digital processed value.
DPVFail	BOOL	Digital processed value inoperable	Digital processed value inoperable.
HFail	BOOL	Hardware inoperable	Digital input hardware inoperable. This is a trip alarm based on the AlmType selected.

The following table describes the input/output parameter of the DigIn block:

Parameter	Type	Description
Stx	DigIn_St	Status and control of the DigIn block. Refer to the Structure to interface with the HMI *(see page 196)*.

DigIn - Representation as SCoD in UAG

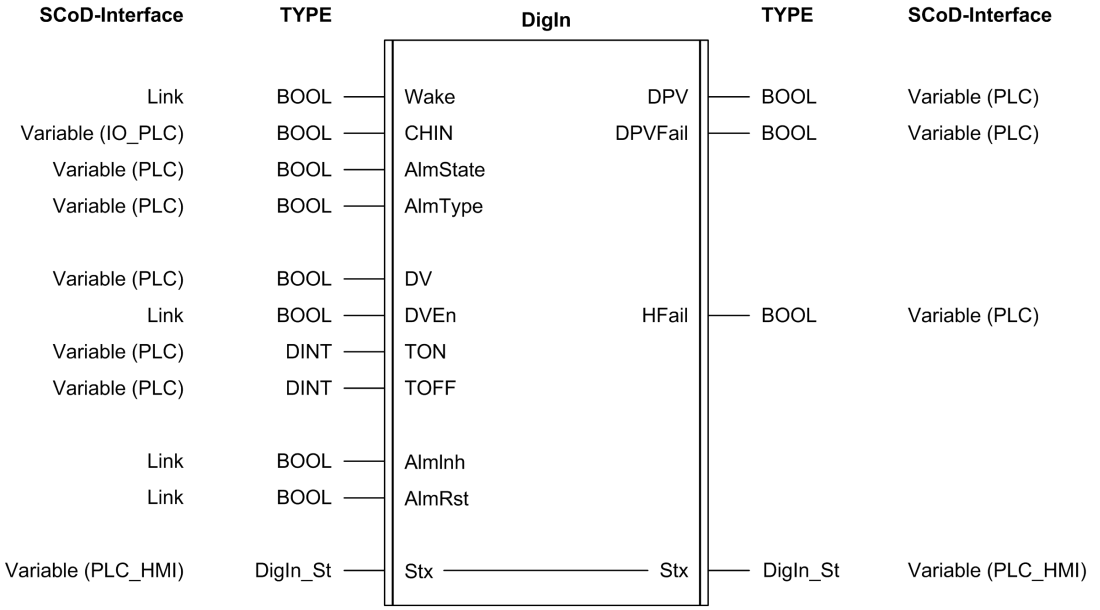
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the `Stx` variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type `HMI` is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the `DigIn` block.



SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Default Value	`DV`	0	**Configuration**	1	Sets the default value
Alarm State	`AlmState`	0	**Configuration**	2	Alarm state

Name	Variable	Initial Value	Property Tab	Position	Description
Timer ON (msec)	TON	0	**Configuration**	3	Timer ON
Alarm Type	AlmType	0	**Configuration**	8	Alarm type
Timer OFF (msec)	TOFF	0	**Configuration**	9	Timer OFF
Area Information	AreaInfo	Not available	**Device Data**	1	Area information
Asset ID	AssetId	Not available	**Device Data**	7	Asset ID

DigIn - Representation in the HMI

Digital Input (Genie)

The DigIn block is represented in the HMI by the following genie.

Genie Elements

Symbol	Meaning
	Opens the DigIn popup.
E, F, H, I, J, L, P, S, T, V, W, X and Z	Different digital indications are represented, that is, voltage, flow, current, hand, power, level, pressure, speed, temperature, vibration, weight, miscellaneous and limit switch respectively.
	The instance name of the DigIn block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

Animation Description of the Genie

Symbol	Animation	Meaning
	Line color: Light gray Fill color: Light gray	Digital input block is in Sleep mode.
	Line color: Dark gray Fill color: Light gray	Digital indicator is awake and indicates 0 without alarms.
	Line color: Dark gray Fill color: Green	Digital indicator is awake and indicates 1 without alarms.
	Line color: Flashing yellow Fill color: Dark gray	Alert alarm in the digital input block is active.
	Line color: Steady yellow Fill color: Dark gray	Alert alarm in the digital input block is acknowledged.
	Line color: Dark gray Fill color: Flashing red	Trip alarm in the digital input block is active.

Symbol	Animation	Meaning
	Line color: Dark gray Fill color: Steady red	Trip alarm in the digital input block is acknowledged.
	Line color: Flashing purple	Hardware inoperable alarm in the digital input block is active.
	Line color: Steady purple	Hardware inoperable alarm in the digital input block is acknowledged.

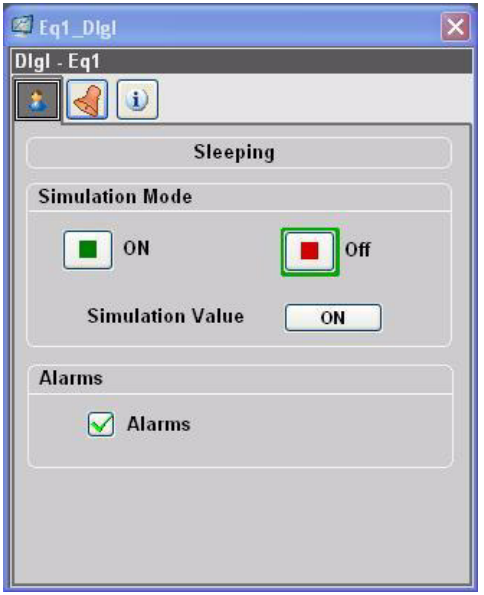
Item	Meaning
S	S is displayed on the left top corner when the simulator is active.
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Popup

The `DigIn` block popup has the following tabs:

- **Operator** tab
- **Alarm** tab
- **Information** tab

Operator Tab



This tab has 3 sub-sections:

- **Status**
- **Simulation Mode**
- **Alarms**

The sub-section **Status** displays the current status.

The possible displays are listed as follows:

- **Sleeping**
- **OFF**
- **ON**
- **Alarm**
- **Warning**
- **Hardware Failure**

The following table describes the sub-section **Simulation Mode**:

Item	Description
ON	Switch on the simulation.
Off	Switch off the simulation.
Simulation Value	Enter the simulation value or set point.

The following table describes the sub-section **Alarms**:

Item	Description
Alarms	Select the check box to enable the alarms. Click to clear the check box to disable the alarms.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

DigIn_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type DigIn_St for the DigIn block:

Element	Type	Description
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI
Dummy	WORD	Not used

Bit description for the Ctw element:

Bit	Description
0	Simulation selection
1	Simulation value
2	Alarm enable
3 ...6	Not used
7	Reset
8...15	Not used

Bit description for the Stw element:

Bit	Description
0	Wake-up
1	Channel input
2	Alarm state
3	Alarm type
4	Digital processed value
5	Alarm inhibition
6	Digital processed value fail
7 ... 15	Not used

Bit description for the `Alw` element:

Bit	Description
0	Hardware inoperable
1	Digital alarm
2	Digital alert
3...15	Not used

8.6 DigOut: Digital Output

Overview

This section describes the DigOut block.

What's in this Section?

This section contains the following topics:

Topic	Page
DigOut - Description	199
DigOut - Representation as SCoD in UAG	202
DigOut - Representation in the HMI	203
DigOut_St Derived Data Type	206

DigOut - Description

General

The DigOut block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB DigOut
Vijeo Citect	Genie DigOut
UAG	SCoD DigOut

Function Description

The DigOut block is used to read digital inputs from the set point and sends the digital processed value to digital output modules.

A wake-up command is given by the GrpCtrl block to wake-up the digital output block from Sleep mode. The digital set point comes from an external block.

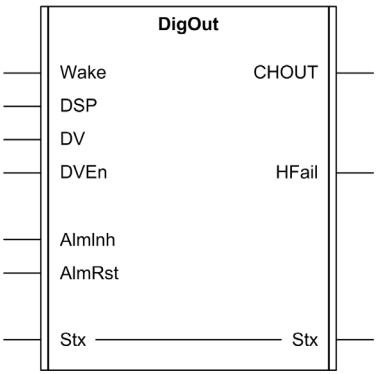
The set point value is processed and a digital output (CHOUT) is given to the respective channel of the digital output module. When the simulation is switched on, the simulation value is set from the popup is considered as the output (CHOUT).

The alert and trip are indicated in the SCADA. The number of active alarms is shown in the popup of digital output block. The details of the active alarms are shown in the **Alarm** tab of the popup. Inoperable hardware alarms and indications are obtained.

The alarms for digital input block can be inhibited by connecting the logic to the input pin (AlmInh). The input pin (AlmRst) connected from GrpCtrl block or the operator from the popup resets the trip alarms.

Representation in FBD

The following figure shows the FBD representation of the DigOut block.



Parameter Description

The following table describes the input parameters of the DigOut block:

Parameter	Type	Description	Details
Wake	BOOL	Group wake-up	Group wake-up and request for group availability check: • Wake = 0: Group is in Sleep mode. • Wake = 1: Group is active. This input is connected from output Wake of GrpCtrl or RouteMgt.
DSP	BOOL	Digital set point	Digital set point.
DV	BOOL	Default value	Default value.
DVEn	BOOL	Enable the default Value, for example, inoperable channel	Enable the default value, for example, inoperable channel.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

The AlmInh parameter to inhibit alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the DigOut block:

Parameter	Type	Description	Details
AlmInh	BOOL	Alarm inhibition	• AlmInh = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • AlmInh = 0: Alarms are not inhibited.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

To reset, alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the DigOut block:

Parameter	Type	Description	Details
AlmRst	BOOL	Reset	• AlmRst = 1: Resets all the alarms generated by the block. • AlmRst = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from GRst pin of GrpCtrl block and resets the trip alarm.

The following table describes the output parameters of the DigOut block:

Parameter	Type	Description	Details
CHOUT	BOOL	Channel Output	Calculated value of the output.
HFail	BOOL	Hardware inoperable	Digital output inoperable hardware.

The following table describes the input/output parameter of the DigOut block:

Parameter	Type	Description
Stx	DigOut_St	Status and control of the DigOut block. Refer to the Structure to interface with the HMI *(see page 206)*

DigOut - Representation as SCoD in UAG

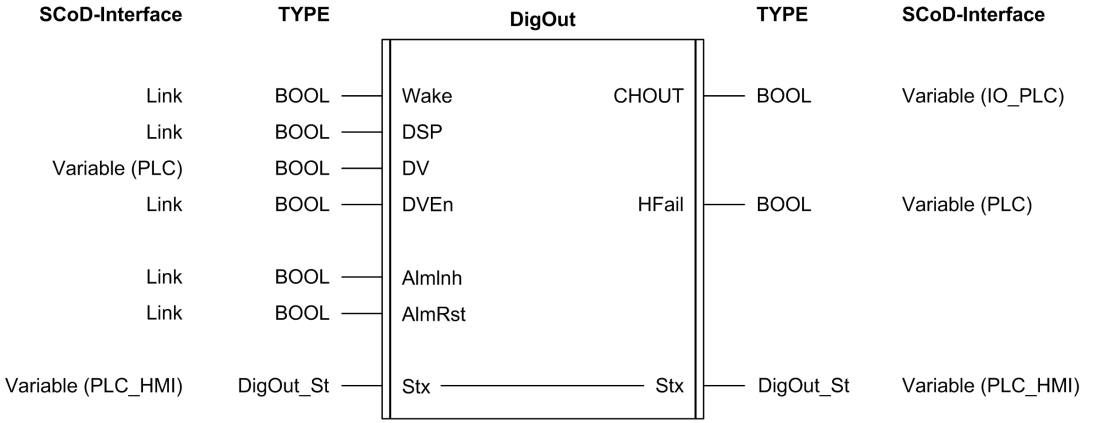
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the Stx variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type HMI is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the DigOut block.



SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Default value	DV	0	**Configuration**	1	Sets the default value
Area Information	AreaInfo	Not available	**Device Data**	1	Area information
Asset ID	AssetId	Not available	**Device Data**	7	Asset ID

DigOut - Representation in the HMI

Digital Output (Genie)

The Digout block is represented in the HMI by the following genie.

Genie Elements

Symbol	Meaning
	Opens the DigOut popup.
E, F, H, I, J, L, P, S, T, V, W, X and Z	Different digital indications are represented, that is, voltage, flow, current, hand, power, level, pressure, speed, temperature, vibration, weight, miscellaneous, and limit switch respectively.
	The instance name of the DigOut block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

Animation Description of the Genie

Symbol	Animation	Meaning
	Line color: Light gray Fill color: Light gray	Digital output block is in Sleep mode.
	Line color: Dark gray Fill color: Light gray	Digital output awake and given value 0 without alarms.
	Line color: Dark gray Fill color: Green	Digital output awake and given value 1 without alarms.
	Line color: Flashing purple	Hardware inoperable alarm in the digital output block is active.
	Line color: Steady purple	Hardware inoperable alarm in the digital output block is acknowledged.

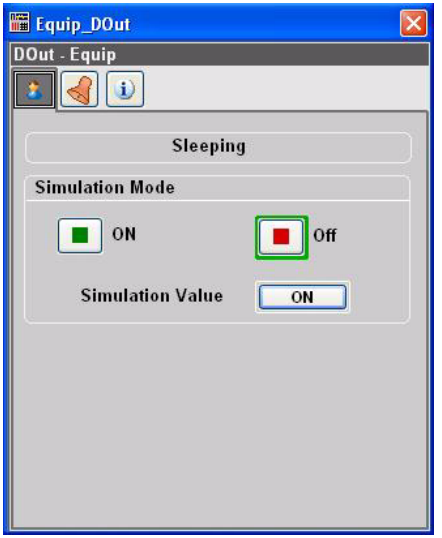
Symbol	Meaning
S	S is displayed on the left top corner when the simulator is active.
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Popup

The `DigOut` block popup has the following tabs:

- **Operator** tab
- **Alarm** tab
- **Information** tab

Operator Tab



This tab has 2 sub-sections:

- **Status**
- **Simulation Mode**

The sub-section **Status** displays the current status.

The possible displays are listed as follows:

- **Sleeping**
- **Hardware Failure**
- **ON**
- **OFF**

The following table describes the sub-section **Simulation Mode**:

Item	Description
ON	Switch on the simulation.
Off	Switch off the simulation.
Simulation Value	Enter the simulation value or set point.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

DigOut_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type DigIn_St for the Digout block:

Element	Type	Description
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI
Dummy	WORD	Not used

Bit description for the Ctw element:

Bit	Description
0	Simulation selection
1	Simulation value
2 ...6	Not used
7	Reset
8...15	Not used

Bit description for the Stw element:

Bit	Description
0	Wake-up
1	Channel output
2 ... 3	Not used
4	Set point
5	Alarm inhibition
6 ... 15	Not used

Bit description for the Alw element:

Bit	Description
0	Hardware inoperable
1...15	Not used

8.7 Mot1dVS: Motor 1 Direction Variable Speed

Overview

This section describes the Mot1dVS block.

What's in this Section?

This section contains the following topics:

Topic	Page
Mot1dVS - Description	208
Mot1dVS - Representation as SCoD in UAG	220
Mot1dVS - Representation in the HMI	223
Mot1dVS_St Derived Data Type	233

Mot1dVS - Description

General

The Mot1dVS block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB Mot1dVS
Vijeo Citect	Genie Mot1dVS
UAG	SCoD Mot1dVS

Function Description

The Mot1dVS block is used to control electrical motors with 1 direction of rotation and variable speed.

This block supports 3 modes of operation:
- Auto mode
- Manual mode
- Local mode

By default, the motor will be in Auto mode. The motor will be forced to Auto or Local mode if the mode selection input pin (AutoSel or LocalSel) is HIGH. Local pin has the highest priority. When the Local mode is selected, the block functions in Local mode irrespective of other mode selections.

If the mode selection input pin (AutoSel or LocalSel) is LOW, operator changes the mode of motor from the popup of motor block. When the motor is under maintenance, the Out of Service mode can be activated from the popup.

Auto Mode Operation

A wake-up command (**Wake**) is given by the GrpCtrl block to wake-up the motor from sleep. A group start-up pulse command (**Start**) is given either from the GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group. This sets the output GActive as HIGH if all the interlocks (StartILck, SftyILck, SeqILck, ProcILck) and all the feedbacks (Lstop, Avail, OverLoad, EStop) are HIGH. This sets the output (OP) as HIGH when the input conditional start (CondStart) is HIGH, and motor starts in the selected direction.

Ready output (Ready) is set HIGH when all interlocks and conditions are HIGH.

Interlock output (ILck) is set HIGH when inputs starting interlock (StartILck), safety interlock (SftyILck), sequential interlock (SeqILck) and process interlock (ProcILck) are set HIGH.

After the start command (**Start**) is issued and output (OP) is set HIGH within the specified run feedback delay, the electrical run feedback (FbRun) has to be obtained. This electrical run feedback is continuously monitored by motor block. If it is not operational, motor is tripped, and an alarm is generated.

After the start command (**Start**) is issued and motor is operational, if run feedback (FbRun) is not available for 500 ms, the motor is tripped, and an alarm is generated.

After the start command (**Start**) is issued and the output (OP) is set HIGH within the specified process feedback delay, the process feedback (FbProc) has to be obtained. This process feedback is continuously monitored by the motor block. If it is not operational, motor is tripped, and an alarm is generated.

After the start command (**Start**) is issued, output (OP) is set HIGH, and motor is operational. If process feedback (FbProc) is not available for specified loss of process feedback delay, the motor is tripped, and an alarm is generated.

NOTE: The process feedback can be bypassed by selecting the **Bypass Process f/b** button from the HMI.

A conditional start can be given to this block when the Conditional Start is enabled. This input is used for connecting process logic to control the motor under normal conditions.

NOTE: Conditional Start, Group Active, Sequential Start and Sequential Stop are applicable in Auto mode only.

The start command (**Start**) is issued, and output (OP) is set HIGH when the run feedback (FbRun) and process feedback (FbProc) are obtained. Run output (Run) is set HIGH, and pulse command (**SeqStart**) is given to the next device to start.

Group stop command (**Stop**) or group quick stop command (**QStop**) sets the output command (**OP**) as LOW. Group stop command (**Stop**) is received by the motor from the GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group. Group quick stop command (**QStop**) is received by the motor from the GrpCtrl block.

The stop command (**Stop**) is issued and output (OP) is set LOW when the run feedback (FbRun) and process feedback (FbProc) are set LOW. Run output (Run) is set LOW and sequential stop command (**SeqStop**) is given to the next device to stop.

The quick stop command (**QStop**) is issued and output (OP) is set LOW and sequential stop command (**SeqStop**) is given to the next device to stop.

Manual Mode Operation

Run and **Stop** commands are issued by the operator from the popup of the motor block. Run and process feedback monitoring are applicable as in Auto mode. The operator sets the speed of the motor from the popup and the same is sent as output (SP) that will be connected to drive block. The process value is read from the drive block and is shown in the popup.

NOTE: Sequential interlock is not applicable in Manual mode.

Local Mode Operation

Stage	Description
1	Start-up pulse command (**LStart**) comes from the local control panel of the motor that sets the output `OP` HIGH and motor starts in the selected direction.
2	Stop command (**LStop**) also comes from the local control panel of the motor. This input always must be HIGH for the motor to be started and is applicable in all the modes of operation. **NOTE:** This serves as an **Emergency Stop** command.

Run and Process feedback monitoring are applicable as in Auto mode.

Interlock Types

There are 4 types of interlocks. The result of individual interlock condition blocks is connected to these interlock pins:

Interlock Name	Parameter	Description
Starting Interlock	`StartILck`	These interlocks are valid before motor start-up only.
Safety Interlock	`SftyILck`	All the safety conditions are taken into consideration. This input is active in all the modes.
Sequential Interlock	`SeqILck`	All the sequential conditions are taken into consideration. This input is active only in Auto mode.
Process Interlock	`ProcILck`	All the process conditions are taken into consideration. This input is active in Auto and Manual mode.

Feedback

The status of the interlock is displayed in the popup. Detailed description of the interlocks is displayed in the **Interlock** tab of the popup.

Local stop signal (`LStop`) is connected from stop push button of local panel and is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated in Auto and Manual modes only.

The availability (`Avail`) of the motor is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated. The overload feedback (`OverLoad`) of the motor is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated.

An emergency stop signal (`EStop`) comes from MCC panel emergency shut-down push button and is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated.

The alert and trip are indicated in the SCADA. The number of active alarms is shown in the popup of the motor. The details of the active alarms are shown in the **Alarm** tab of the popup. Based on configured conditions, the alarms for this block can be inhibited by connecting the logic to the input pin `AlmInh`. The input pin `AlmRst` that is connected from `GrpCtrl` block or the operator from popup can reset the trip alarms.

Configuration

The load and energy values are input to the motor block and these values are displayed in the popups of the motor block.

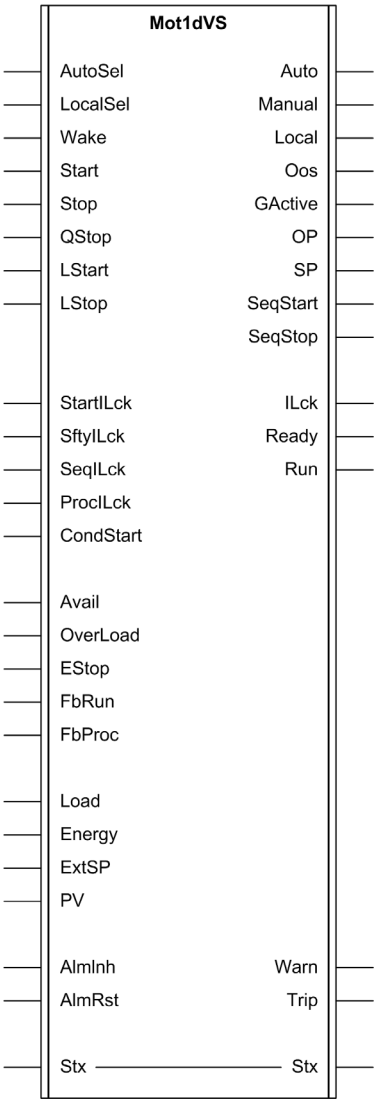
The run hours of the motor is displayed in the **Configuration** tab of the popup. After the maintenance of the motor, the run hours can be reset from the popup.

NOTE: The following parameters in the **Configuration** tab have user restriction from the HMI:

- **Run feedback delay**
- **Process feedback delay**
- **Loss of Process feedback delay**
- **Run hours R** (reset)

Representation in FBD

The following figure shows the FBD representation of the `Mot1dVS` block.



Parameter Description

The following table describes the input parameter of the Mot1dVS block:

Parameter	Type	Description	Details
AutoSel	BOOL	Auto mode selection	Selection of operation mode (Auto mode): ● AutoSel = 1: Block is forced to Auto mode. Mode selection option in HMI is grayed out. ● AutoSel = 0: Block is set free for operator to select the mode from HMI.
LocalSel	BOOL	Local mode selection	Selection of operation mode (Local mode): ● LocalSel = 1: Block is forced to Local mode. Mode selection option in HMI is grayed out. ● LocalSel = 0: Block is set free for operator to select the mode from HMI.
Wake	BOOL	Group wake-up	Group wake-up and request for group availability check: ● Wake = 1: Group is active. ● Wake = 0: Group is in Sleep mode. This input is connected from output GWake of GrpCtrl.
Start	BOOL	Group start	**Start** command to the motor. When Start1 = 1: OP pin is set to 1, this is applicable in Auto mode only. This input is connected from the GStart pin of GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group.
Stop	BOOL	Group stop	**Stop** command to the motor. When Stop = 1: OP pin is set to 0. Applicable in Auto mode only. This input is connected from the GStop pin of GrpCtrl or GrpStpMgt or RouteMgt or previous device in the group.
QStop	BOOL	Group quick stop	**Stop** command to the motor. When GQStop = 1: OP pin is set to 0. Applicable in Auto mode only. This input is connected from the GQStop pin of GrpCtrl.

Parameter	Type	Description	Details
LStart	BOOL	Local start	**Start** command to the motor. When LStart = 1: OP pin is set to 1. Applicable in Local mode only. This input is connected from the **Start** push button of local IO panel.
LStop	BOOL	Local stop	**Stop** command to the motor. When LStop = 0: OP pin is set to 0. Applicable in Local mode only. This input is connected from the **Stop** push button of local IO panel.
StartILck	BOOL	Starting interlock	• StartILck = 1: The starting interlock conditions are operational. • StartILck = 0: The starting interlock conditions are not operational. Applicable only in Auto and Manual modes of operation. Applicable only before starting of the motor. After motor is started, StartILck is not considered.
SftyILck	BOOL	Safety interlock	• SftyILck = 1: The safety interlock conditions are operational. • SftyILck = 0: The safety interlock conditions are not operational, trips the motor and generates an alarm. Applicable in all the modes of operation (Auto, Manual, and Local).
SeqILck	BOOL	Sequential interlock	• SeqILck = 1: The sequential interlock conditions are operational. • SeqILck = 0: The sequential interlock conditions are not operational, trips the motor and generates an alarm. Applicable in Auto mode only.
ProcILck	BOOL	Process interlock	• ProcILck = 1: The process interlock conditions are operational. • ProcILck = 0: The process interlock conditions are not operational, trips the motor and generates an alarm. Applicable in Auto and Manual modes only.

Parameter	Type	Description	Details
CondStart	BOOL	Conditional start	• CondStart = 1: **Start** command to the motor, OP = 1. • CondStart = 0: **Stop** command to the motor, OP = 0. Applicable in Auto mode only. If GActive = 1, CondStart is used to start or stop a motor based on process condition.
Avail	BOOL	Availability of motor	• Avail = 1: Motor is electrically available. • Avail = 0: Motor is electrically not available. **Stop** command to the motor, OP = 0. This signal comes from the MCC outgoing circuit (electrical protection active, circuit breaker switched off, drawer out, and so on).
OverLoad	BOOL	Overload	• OverLoad = 1: Motor is not in overload condition. • OverLoad = 0: Motor is not in overload condition. **Stop** command to the motor, OP = 0. This signal comes from the MCC outgoing circuit (electrical protection active).
EStop	BOOL	Emergency stop	• EStop = 1: Plant process condition is operational for the motor to run. • EStop = 0: Emergency situation. Plant process condition is not operational for the motor to run. **Stop** command to the motor, OP = 0. Applicable in all operation modes (Auto, Manual, and Local). This input is connected from the **Emergency Stop** push button in the field.

Parameter	Type	Description	Details
FbRun	BOOL	Run feedback of motor - Electrical	Once the motor is given a **Start** command from output OP, its electrical run feedback is continuously monitored. This signal comes from the MCC outgoing circuit. If OP = 1 and FbRun = 0, then **Stop** command to the motor, OP = 0 is given and an alarm is generated after time delay. If OP = 0 and FbRun = 1, an alarm is generated after time delay.
FbProc	BOOL	Run feedback of motor - Process	Once the motor is given a **Start** command from output OP, its electrical run feedback is continuously monitored. This signal comes from Speed monitoring devices, Zero Speed switch, Proximity switches, and so on. If OP = 1 and FbProc = 0, then **Stop** command to the motor, OP = 0 is given and an alarm is generated after time delay. If OP = 0 and FbProc = 1, an alarm is generated after time delay.
Load	REAL	Load value of the motor	Load value of motor in A/kW.
Energy	REAL	Energy value of the motor	Energy value of motor in kWh.
ExtSP	REAL	External set point	External set point.
PV	REAL	Process value	Current running speed in RPM.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

The AlmInh parameter to inhibit alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the Mot1dVS block:

Parameter	Type	Description	Details
AlmInh	BOOL	Alarm inhibition	• AlmInh = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • AlmInh = 0: Alarms are not inhibited.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

To reset, alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameters of the Mot1dVS block:

Parameter	Type	Description	Details
AlmRst	BOOL	Reset	• AlmRst = 1: Resets all the alarms generated by the block. • AlmRst = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from GRst pin of GrpCtrl block and resets the trip alarm.

The following table describes the output parameters of the Mot1dVS block:

Parameter	Type	Description	Details
Auto	BOOL	Auto mode	Auto = 1: Auto mode is selected. Motor operation is group based on GStart and GStop.
Manual	BOOL	Manual mode	Manual = 1: Manual mode is selected. Motor operation is based on commands **Start** and **Stop** given by operator from HMI.
Local	BOOL	Local mode	Local = 1: Local mode is selected. Motor operation is based on commands **LStart** and **LStop** given from field.

Parameter	Type	Description	Details
Oos	BOOL	Out of Service mode	Oos = 1: Out of Service mode is selected. Motor is not operated and OP = 0.
GActive	BOOL	Group active	The motor is started in the group and is active in the group. Start command pulse **Start** is obtained, GActive = 1.
OP	BOOL	Output command to motor.	• OP = 1: Motor is given command to start. • OP = 0: Motor is given command to stop. The output switches on the contactor or circuit breaker controlling the motor in forward direction.
SP	REAL	Set point	Speeds set point for the motor. Generally connected to variable speed drive block.
SeqStart	BOOL	Start command to next motor in group.	Start command (pulse) is received and outputs OP and Run = 1, sequential start command pulse is given to the next motor connected in the group.
SeqStop	BOOL	Stop command to next motor in group.	Stop or QStop = 1 and OP and Run = 0, sequential stop command is given to the next motor connected in the group.
ILck	REAL	Interlock status	• ILck = 1: All the interlocks are operational. • ILck = 0: All the interlocks are not operational. There is an active interlock present.
Ready	BOOL	Motor ready to start.	• Ready = 1: All the conditions to start the motor are operational. • Ready = 0: The conditions to start the motor are not operational. Motor will not be given **Start** command.
Run	BOOL	Motor running	Motor is running. This output is given if the motor is running and run feedback (FbRun) is received. Applicable in all the operation modes.

Parameter	Type	Description	Details
Warn	BOOL	Alert	• Warn = 1: The motor is operational with no active alert present. • Warn = 0: The motor is operational with an active alert present.
Trip	BOOL	Trip	• Trip = 1: The motor is operational with no active trip alarm present. • Trip = 0: The motor has tripped with an active trip alarm present.

The following table describes the input/output parameter of the Mot1dVS block:

Parameter	Type	Description
Stx	Mot1dVS_St	Status and control of the motor 1 direction variable speed block. Refer to the Structure to interface with the HMI *(see page 233)*.

Mot1dVS - Representation as SCoD in UAG

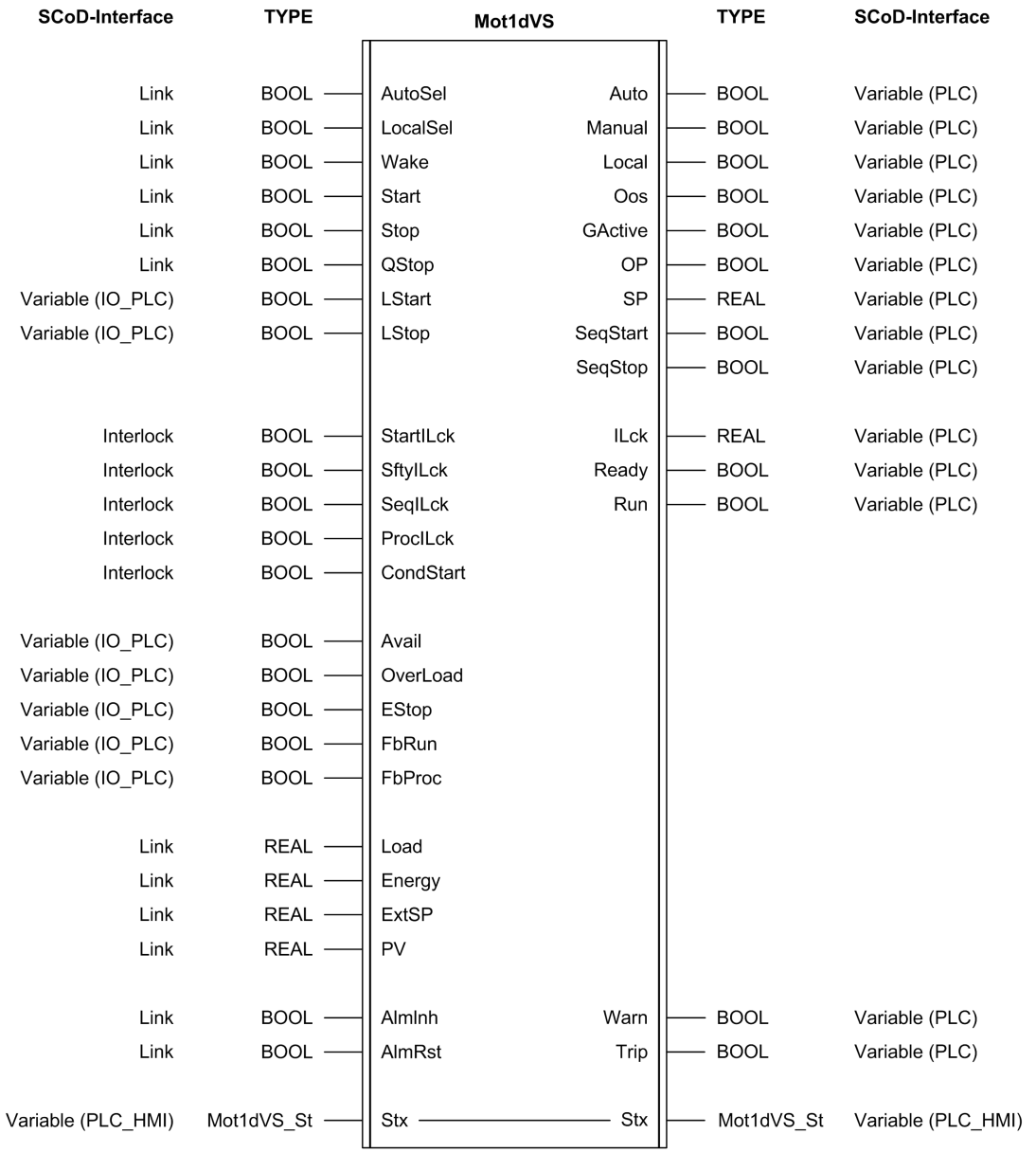
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the `Stx` variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type `HMI` is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the `Mot1dVS` block.



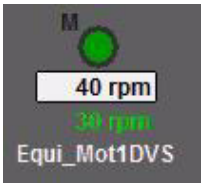
SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Display format (time)	Stx.RunHr	#####	**Configuration**	1	Run hours of motor.
Run fb delay (Sec)	Stx.TRunFb	0	**Configuration**	2	Sets run feedback delay (s).
Loss of process fb delay (sec)	Stx.TLProcFb	0	**Configuration**	3	Sets loss of process feedback delay (s).
Load measurement unit	Stx.Load	a	**Configuration**	4	Load of motor (A/kW).
Display format (SP)	Stx.SP	###.##	**Configuration**	7	Set point (RPM).
Process fb delay (Sec)	Stx.TProcFb	0	**Configuration**	8	Sets process feedback delay (s).
Ramp delay (sec)	Stx.TRamp	0	**Configuration**	9	Ramp delay
Energy measurement unit	Stx.Energy	Not available	**Configuration**	10	Energy of motor (kWh).
Low limit value	Stx.SPL	0.0	**Limits**	1	Set point low limit.
High limit value	Stx.SPH	100.0	**Limits**	8	Set point high limit.
Load Eng. Min	Stx.Load	0	**Range**	1	Sets the minimum load Eng. value.
Energy Eng. Min	Stx.Energy	0	**Range**	2	Sets the minimum energy Eng. value.
Eng. Min	Stx.SP	0.0	**Range**	3	Sets the minimum Eng. value.
Eng.Max	Stx.Load	100.0	**Range**	9	Sets the maximum Eng. value.
Load Eng. Max	Stx.Load	100.0	**Range**	7	Sets the maximum load Eng. value.
Energy Eng.Max	Stx.Energy	100.0	**Range**	8	Sets the maximum energy Eng. value.
Area Information	AreaInfo	Not available	**Device Data**	1	Area information
Asset ID	AssetId	Not available	**Device Data**	7	Asset ID

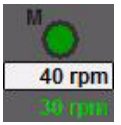
Mot1dVS - Representation in the HMI

Motor 1 Direction Variable Speed (Genie)

The Mot1dVS block is represented in the HMI by the following genie.



Genie Elements

Symbol	Meaning
	Opens the Mot1dVS popup.
	Set point value of the motor 1 direction variable speed is displayed.
	Process value of the motor 1 direction variable speed is displayed.
	The instance name of the Mot1dVS block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

Animation Description of the Genie

Symbol	Animation	Meaning
	Line color: Steady light gray Fill color: Steady light gray	Motor is in Sleep mode.
	Line color: Steady dark gray Fill color: Steady white	• Motor is awake or in Local mode or Manual mode and ready. • Motor is awake or in Local mode or Manual mode and ready after alarms are reset.
	Line color: Dark gray Fill color: Light gray	Motor is awake or in Local mode or Manual mode and not ready.
	Line color: Steady dark gray Fill color: Flashing green	Motor is started and feedback is yet to be obtained or motor is stopping.

Symbol	Animation	Meaning
	Line color: Steady dark gray Fill color: Steady green	Motor is running.
	Line color: Flashing yellow Fill color: Steady green	Motor is running, and an alert alarm is active.
	Line color: Steady yellow Fill color: Steady green	Motor is running, and an alert alarm is acknowledged.
	Line color: Steady dark gray Fill color: Flashing red	Motor is tripped, and a trip alarm is active.
	Line color: Steady dark gray Fill color: Steady red	Motor is tripped, and a trip alarm is acknowledged.

Modes

On the top left corner of the motor 1 direction variable speed genie, the mode of operation is indicated.

The following table describes the mode of operations:

Mode	Meaning
A	Motor is operating in the Auto mode.
M	Motor is operating in the Manual mode.
L	Motor is operating in the Local mode.
O	Motor is in Out of Service mode.

Symbol	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Alarm Masking

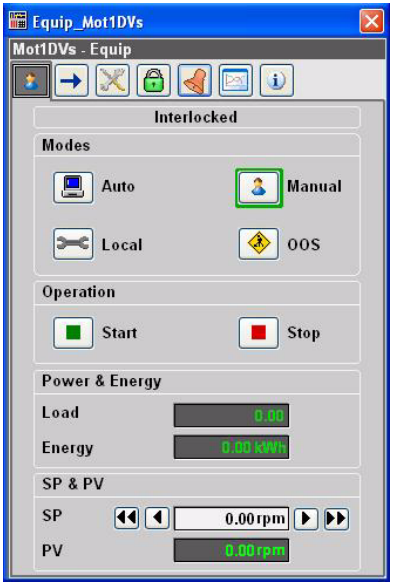
Symbol	Meaning
	Trip or alert alarms in the motor 1 direction variable speed is masked.

Popup

The `Mot1dVS` block popup has the following tabs:

- **Operator** tab
- **Extended** tab
- **Configuration** tab
- **Interlock** tab
- **Alarm** tab
- **Trend** tab
- **Information** tab

Operator Tab



This tab has 5 sub-sections:

- **Status**
- **Modes**
- **Operation**
- **Power & Energy**
- **SP & PV**

The sub-section **Status** displays the current status.

The possible displays are listed as follows:

- **Sleeping**
- **Group Active**
- **Ready**
- **Not Ready**

- **Starting**
- **Stopping**
- **Running**
- **Interlocked**
- **Warning**
- **Trip**

The following table describes the sub-section **Modes**:

Item	Description
Auto	Switches to Auto mode controlled by the PLC program.
Manual	Switches to Manual operation mode controlled by the HMI.
Local	Switches to Local operation mode controlled by the local control box of the motor.
OOS	Switches to Out of Service mode for maintenance purpose of the object (applicable in all modes).

The following table describes the sub-section **Operation**:

Item	Description
Start	**Start** command is given to motor in Manual mode only.
Stop	**Stop** command is given to motor in Manual mode only.

The following table describes the sub-section **Power & Energy**:

Item	Description
Load	Displays load of motor in A/kW.
Energy	Displays energy consumed by motor in kWh/MWh.

The following table describes the sub-section **SP & PV**:

Item	Description
SP	Displays set point value (SP) of motor in RPM.
PV	Displays process value (PV) of motor in RPM.

Symbol	Description
	Set point value is ramped down by 1 step.
	Set point value is ramped down to low limit.

Symbol	Description
	Set point value is ramped up by 1 step.
	Set point value is ramped up to high limit.

Extended Tab

This tab has 3 sub-sections:

- **Feedback**
- **Mask**
- **Bypass**

The following table describes the sub-section **Feedback**:

Item	Animation	Description
Local Stop	Steady green	Motor locally stopped feedback is obtained.
	Steady gray	Motor locally not stopped feedback is obtained.
Conditional Start	Steady green	Motor is started.
	Steady gray	Motor is not started.

Item	Animation	Description
Available	Steady green	Motor electrically available feedback is obtained.
	Steady gray	Motor electrically not available feedback is obtained.
Over Load	Steady green	Motor overload feedback is obtained.
	Steady gray	Motor overload feedback is not obtained.
Emergency Shutdown	Steady green	Emergency stop feedback is obtained.
	Steady gray	Emergency stop feedback is not obtained.
Run	Steady green	Motor running feedback is obtained.
	Steady gray	Motor not running feedback is obtained.
Process	Steady green	Process feedback is obtained.
	Steady gray	Process feedback is not obtained.

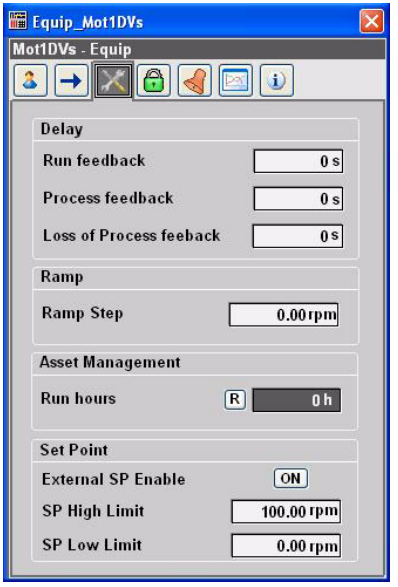
The following table describes the sub-section **Mask**:

Item	Description
Alarm Mask	Masks the indicated current trip alarms of this block in the alarm list.
Warning Mask	Masks the current alerts of this block in the alarm list.

The following table describes the sub-section **Bypass**:

Item	Description
Process feedback	Bypasses the process feedback.

Configuration Tab



This tab has 3 sub-sections:

- **Delay**
- **Ramp**
- **Asset Management**
- **Set Point**

The following table describes the sub-section **Delay**:

Item	Description
Run feedback	Displays run feedback delay of motor, configurable from HMI. After starting motor, if running feedback is not received before delay time, an alarm is generated, and the motor is tripped.
Process feedback	Displays process feedback delay of motor, configurable from HMI. After starting motor, if process feedback is not received before delay time, an alarm is generated, and the motor is tripped.
Loss of Process feedback	Displays loss of process feedback delay of motor, configurable from HMI. After receiving process feedback, if there is loss of feedback while motor is running and is not received before delay time, an alarm is generated, and the motor is tripped.

The following table describes the sub-section **Ramp**:

Item	Description
Ramp Step	Configurable and displays the ramp step value.

The following table describes the sub-section **Asset Management**:

Item	Description
Run hours	Displays the run hours of the motor.
	Resets the run hours value of the motor to 0.

The following table describes the sub-section **Set Point**:

Item	Description
External SP Enable	Enables the external set point.
SP High Limit	Configurable and displays the set point high limit.
SP LOW Limit	Configurable and displays the set point low limit.

Interlock Tab

This tab has a sub-section called as **Interlock**, which is described in the following table:

Item	Animation	Description
Starting Interlock	Steady green	**Starting Interlock** is active, and the related device is not blocked.
	Steady gray	**Starting Interlock** is not active, and the related device is blocked.
Safety Interlock	Steady green	**Safety Interlock** is active, and the related device is not blocked.
	Steady gray	**Safety Interlock** is not active, and the related device is blocked.
Sequential Interlock	Steady green	**Sequential Interlock** is active, and the related device is not blocked.
	Steady gray	**Sequential Interlock** is not active, and the related device is blocked.
Process Interlock	Steady green	**Process Interlock** is active, and the related device is not blocked.
	Steady gray	**Process Interlock** is not active, and the related device is blocked.

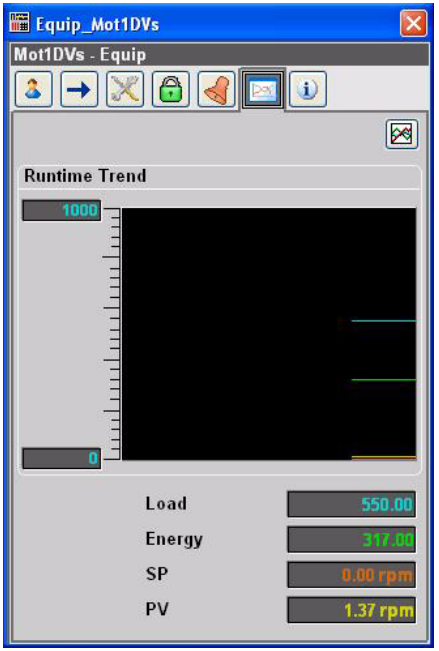
This tab has a sub-tabs Interlock, **STI**, **SI**, **SQI**, and **PI**.

For description about **Interlock** sub-tabs, refer to Commonly Used Tabs *(see page 53)*.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Trend Tab



The sub-section **Runtime Trend** displays the run time trend of the parameters in the control module.

Item	Description
Load	Displays the run time trend of load.
Energy	Displays the run time trend of energy.
SP	Displays the run time trend of set point.
PV	Displays the run time trend of process variable.

Symbol	Meaning
	Displays the historical trend page.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

Mot1dVS_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type Mot1dVS_St for the Mot1dVS block:

Element	Type	Description
Load	REAL	Load of motor in A/kW
Energy	REAL	Energy consumed by motor in kWh/MWh
ExtSP	REAL	External mode set point (RPM)
ManSP	REAL	Manual mode set point (RPM)
PV	REAL	Process value (RPM)
SP	REAL	Set point (RPM)
SPH	REAL	Set point high limit (RPM)
SPL	REAL	Set point low limit (RPM)
Ramp	REAL	Ramp value
RunHr	INT	Run hours of motor
TRunFb	INT	Run feedback delay
TProcFb	INT	Process feedback delay
TLProcFb	INT	Loss of process feedback delay
TRamp	INT	Ramp delay
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word1 to interact with HMI
Stw2	WORD	Status word2 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI
ExtSPEn	BOOL	External set point enable

Bit description for the Ctw element:

Bit	Description
0	Auto mode selection from HMI
1	Manual mode selection from HMI
2	Local mode selection from HMI
3	Out of Service mode selection from HMI
4	Start command from HMI
5	Not used
6	Stop command from HMI

Bit	Description
7	Alarm reset from HMI
8	Run hour reset from HMI
9	Bypass process feedback from HMI
10	Ramp up
11	Ramp down
12	Ramp maximum
13	Ramp minimum
14	Alert mask
15	Alarm mask

Bit description for the `Stw1` element:

Bit	Description
0	Auto mode selection from HMI
1	Local mode selection from HMI
2	Group wake
3	Local stop
4	Starting interlock
5	Safety interlock
6	Sequential interlock
7	Process interlock
8	Motor electrically available
9	Motor overload
10	Emergency stop
11	Run feedback
12	Not used
13	Process feedback
14	Alarm inhibition
15	Not used

Bit description for the `Stw2` element:

Bit	Description
0	Auto mode active status
1	Manual mode active status
2	Local mode active status

Bit	Description
3	Out of Service mode active status
4	Output command active
5	Not used
6	Motor ready status
7	Motor running status
8...9	Not used
10	Conditional start active status
11	Not used
12	Interlock active status
13	Group active status
14	Alert active status
15	Trip active status

Bit description for the `Alw1` element:

Bit	Description
0	Motor locally stop
1	Safety interlock not available
2	Sequential interlock not available
3	Process interlock not available
4	Motor electrically not available
5	Motor overload
6	Motor emergency stop
7	Run feedback not active when motor is running
8	Run feedback active when motor is not running
9...10	Not used
11	Process feedback not active when motor is running
12	Process feedback active when motor is not running
13	Not used
14	Run hours maximum value reached
15	Active trip alarm

8.8 DigAct: Digital Actuator

Overview

This section describes the DigAct block.

What's in this Section?

This section contains the following topics:

Topic	Page
DigAct - Description	237
DigAct - Representation as SCoD in UAG	246
DigAct - Representation in the HMI	249
DigAct_St Derived Data Type	257

DigAct - Description

General

The DigAct block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB DigAct
Vijeo Citect	Genie DigAct
UAG	SCoD DigAct

Function Description

The DigAct block is used to control pneumatic/electrical/electro-pneumatic actuators without electric digital actuator, having 2 defined positions. This control is usually performed through auxiliary solenoid valves fed from an ACC.

Controlled actuators can be either of mono or bi-stable type and input Typ is to be adjusted accordingly.

This block supports 2 modes of operation:
- Auto mode
- Manual mode

By default, the digital actuator will be in Auto mode. The digital actuator is forced to Auto if the mode selection input pin (AutoSel) is HIGH.

If the mode selection input pin (AutoSel) is LOW, operator changes the mode of digital actuator from the popup of digital actuator block. When the digital actuator is under maintenance, the Out of Service mode can be activated from the popup.

Auto Mode Operation

A wake-up command (**Wake**) is given by the GrpCtrl block to wake-up the digital actuator from sleep. A start-up pulse command (**Start**) to achieve position is given either from the GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group.

The actuator is configured as:
- monostable if Typ = 0
- bistable if Typ = 1

If Typ is 0, this sets the outputOP1 as high and once FbPos1 is received, actuator starts to position1.

If Typ is 1 and Direction is 0, this sets the output OP1 as HIGH and actuator starts to position1.

If Typ and Direction are 1, this sets the output OP2 as HIGH and actuator starts to position2.

After the start command (**OP1**) or (**OP2**) is issued to the digital actuator within the specified position feedback delay, the position feedback has to be obtained.

If the position feedback is not obtained, the digital actuator is tripped and an alarm is generated. If the position feedback is obtained, the digital actuator stops.

Direction change of the digital actuator demands stoppage of the digital actuator.

`Defaultpos` is stop position which defines the position at which actuator must stop when **Stop** command is received. If `DefaultPos` is 1, stop position is position1and if `DefaultPos` is 0, stop position is position 2.

The block sets an output group active (`GActive`) if the motor is started in Auto mode. If the block is group active and actuator has reached desired position, a pulse command (**SeqStart**) is given to the next motor to start.

Group stop command (**Stop**) is received by the digital actuator from the `GrpCtrl` or `GrpStrt` or `RouteMgt` or previous device in the group.

Group stop command (**Stop**) or group quick stop command (**QStop**) sets the actuator to stop position, that is, it starts the digital actuator in the stop position (`DefaultPOS`) direction, and then stops.

Group quick stop command (**QStop**) is received by the digital actuator from the `GrpCtrl` block. If the block is group shut down, an output command (**SeqStop**) is given to the next digital actuator to stop.

Manual Mode Operation

The operator initiates the **Position1** and **Position2** commands from the popup of the digital actuator block. Position feedback monitoring is applicable as in Auto mode

Interlock Types

There are 4 types of interlocks. The result of individual interlock condition blocks is connected to these interlock pins:

Interlock Name	Parameter	Description
Starting Interlock	`StartILck`	These interlocks are valid before motor start-up only.
Safety Interlock	`SftyILck`	All the safety conditions are taken into consideration. This input is active in all the modes.
Sequential Interlock	`SeqILck`	All the sequential conditions are taken into consideration. This input is active only in Auto mode.
Process Interlock	`ProcILck`	All the process conditions are taken into consideration. This input is active in Auto and Manual mode.

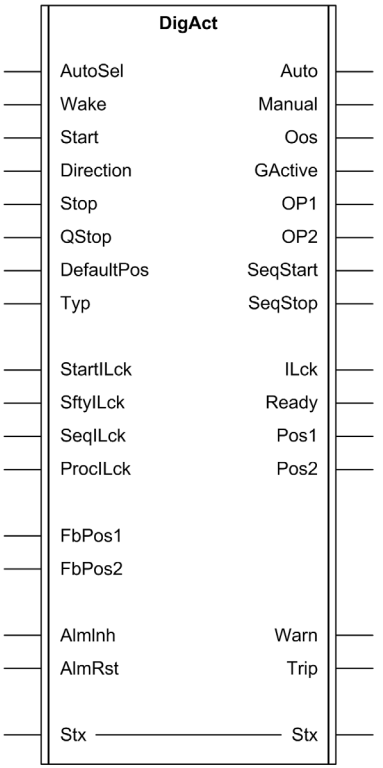
Feedback

The status of the interlock is displayed in the popup. Detailed description of the interlocks is displayed in the **Interlock** tab of the popup.

The alert and trip are indicated in the SCADA. The number of active alarms is shown in the popup of the motor. The details of the active alarms are shown in the **Alarm** tab of the popup. Based on configured conditions, the alarms for this block can be inhibited by connecting the logic to the input pin `AlmInh`. The input pin `AlmRst` that is connected from `GrpCtrl` block or the operator from popup can reset the trip alarms.

Representation in FBD

The following figure shows the FBD representation of the `DigAct` block.



Parameter Description

The following table describes the input parameters of the DigAct block:

Parameter	Type	Description	Details
AutoSel	BOOL	Auto mode selection	Selection of operation mode (Auto mode): • AutoSel = 1: Block is forced to Auto mode. Mode selection option in HMI is grayed out. • AutoSel = 0: Block is set free for operator to select the mode from HMI.
Wake	BOOL	Group wake-up	Group wake-up and request for group availability check: • Wake = 1: Group is active. • Wake = 0: Group is in Sleep mode. This input is connected from output GWake of GrpCtrl.
Start	BOOL	Group start	**Start** command to the actuator. Start = 1: OP pin is set to 1, this is applicable in Auto mode only. This input is connected from the GStart pin of GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group.
Direction	BOOL	Direction	• Direction = 0: Actuator position is closed (Pos1). • Direction = 1: Actuator position is opened (Pos2).
Stop	BOOL	Group stop	**Stop** command to the actuator. Stop = 1: OP1 or OP2 is set to 1, depending on default position input (DefaultPos). Once feedback position is obtained (FbPos1 or FbPos2), OP1 or OP2 is set to 0. This input is connected from the GStop pin of GrpCtrl or GrpStpMgt or RouteMgt or previous device in the group. This is applicable in Auto mode only.

Parameter	Type	Description	Details
QStop	BOOL	Group quick stop	**QStop** command to the actuator. QStop = 1: OP1 or OP2 is set to 1, depending on default position input (DefaultPos). Once feedback position is obtained (FbPos1 or FbPos2), OP1 or OP2 is set to 0. This input is connected from the GStop pin of GrpCtrl or GrpStpMgt or RouteMgt or previous device in the group. This is applicable in Auto mode only.
DefaultPos	BOOL	Default position	• DefaultPos = 1: The actuator position is opened (Pos2). • DefaultPos = 0: The actuator position is closed (Pos1).
Typ	BOOL	Type	• Typ = 1: Bistable. • Typ = 0: Monostable.
StartILck	BOOL	Starting interlock	• StartILck = 1: The starting interlock conditions are operational. • StartILck = 0: The starting interlock conditions are not operational. Applicable only in Auto and Manual modes of operation. Applicable only before starting of the motor. After motor is started, StartILck is not considered.
SftyILck	BOOL	Safety interlock	• SftyILck = 1: All the safety interlock conditions are operational. • SftyILck = 0: All the safety interlock conditions are not operational, trips the motor and generates an alarm. Applicable in all the modes of operation (Auto, Manual, and Local).
SeqILck	BOOL	Sequential interlock	• SeqILck = 1: All the sequential interlock conditions are operational. • SeqILck = 0: All the sequential interlock conditions are not operational, trips the motor and generates an alarm. Applicable in Auto mode only.

Parameter	Type	Description	Details
ProcILck	BOOL	Process interlock	• ProcILck = 1: All the process interlock conditions are operational. • ProcILck = 0: All the process interlock conditions are not operational, trips the motor and generates an alarm. Applicable in Auto and Manual modes only.
FbPos1	BOOL	Position feedback1	Once the actuator is given a **Start** command in forward direction from output OP1, its position feedback is continuously monitored. If Pos1 = 1: **Stop** command to the actuator, OP1 = 0 is given. If OP1 = 1 and FbPos1 = 0 for more than time delay TFbPos, an alarm is generated.
FbPos2	BOOL	Position feedback2	Once the actuator is given a **Start** command in forward direction from output OP2, its position feedback is continuously monitored. If Pos2 = 1: **Stop** command to the actuator, OP2 = 0 is given. If OP2 = 1 and FbPos2 = 0 for more than time delay TFbPos, an alarm is generated.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

The AlmInh parameter to inhibit alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the DigAct block:

Parameter	Type	Description	Details
AlmInh	BOOL	Alarm inhibition	• AlmInh = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • AlmInh = 0: Alarms are not inhibited.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

To reset, alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the DigAct block:

Parameter	Type	Description	Details
AlmRst	BOOL	Reset	• AlmRst = 1: Resets all the alarms generated by the block. • AlmRst = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from GRst pin of GrpCtrl block and resets the trip alarm.

The following table describes the output parameters of the DigAct block:

Parameter	Type	Description	Details
Auto	BOOL	Auto mode	Auto = 1: Auto mode is selected. Motor operation is group based on GStart and GStop.
Manual	BOOL	Manual mode	Manual = 1: Manual mode is selected. Motor operation is based on commands **Start** and **Stop** given by operator from HMI.
Oos	BOOL	Out of Service mode	Oos = 1: Out of Service mode is selected. Motor is not operated and OP = 0.

Parameter	Type	Description	Details
GActive	BOOL	Group active	The motor is started and is active in the group.
OP1	BOOL	Output command to achieve position1	OP1 = 1: Actuator is given command to close.
OP2	BOOL	Output command to achieve position 2	OP2 = 1: Actuator is given command to open.
SeqStart	BOOL	**Start** command to next device in group	**Start** command (pulse output: SeqStart = 0...1) is given to the next device connected in the group.
SeqStop	BOOL	**Stop** command to next device in group	If Stop = 1, **Stop** command (SeqStop = 1) is given to the next device connected in the group.
ILck	BOOL	Interlock status	• ILck = 1: All the interlocks are operational. • ILck = 0: All the interlocks are not operational.
Ready	BOOL	Actuator ready to start	• Ready = 1: All the conditions to start the actuator are operational. • Ready = 0: All the conditions to start the actuator are not operational. Actuator will not be given **Start** command.
Pos1	BOOL	Actuator in position1	Actuator has reached position1. Applicable in all the modes of operation.
Pos2	BOOL	Actuator in position2	Actuator has reached position2. Applicable in all the modes of operation.
Warn	BOOL	Alert	• Warn = 1: The actuator is operational with no active alert present. • Warn = 0: The actuator is operational with an active alert present.
Trip	BOOL	Trip	• Trip = 1: The actuator is operational with no active trip alarm present. • Trip = 0: The actuator has tripped with an active trip alarm present.

The following table describes the input/output parameter of the DigAct block:

Parameter	Type	Description
Stx	DigAct_St	Status and control of the digital actuator. Refer to the Structure to interface with the HMI *(see page 257)*.

DigAct - Representation as SCoD in UAG

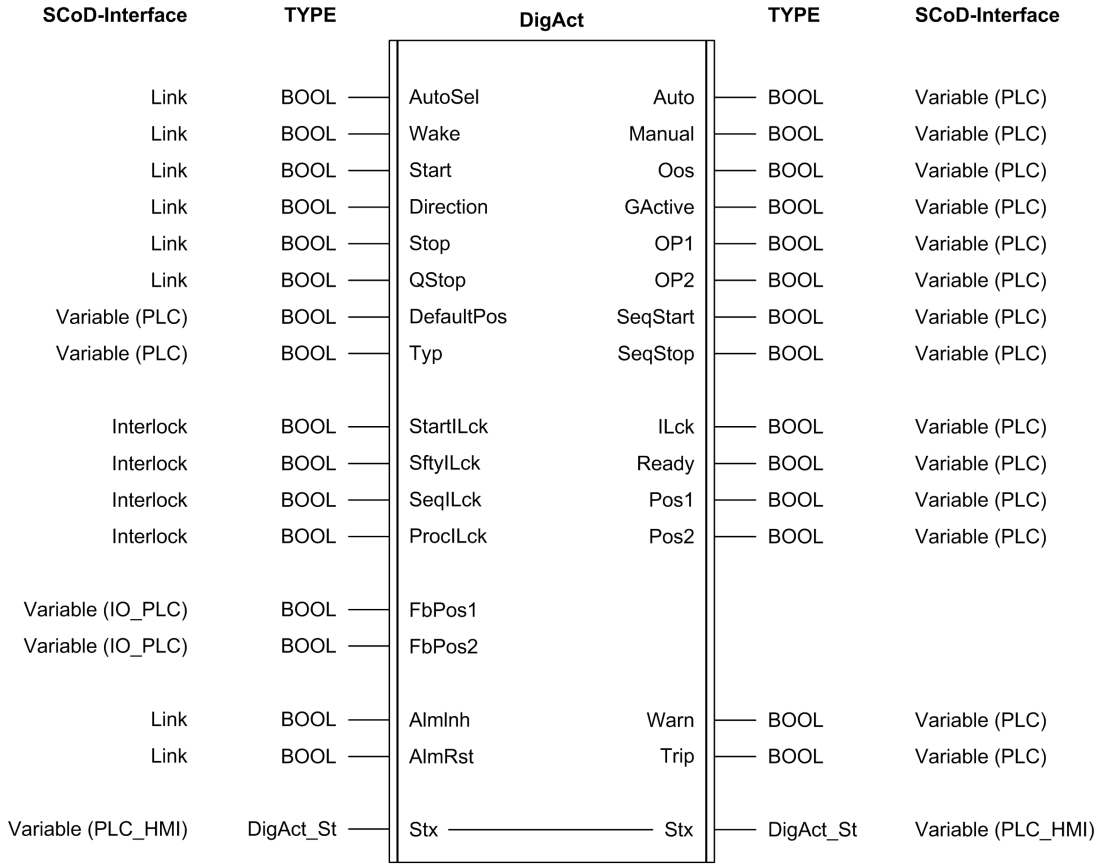
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the Stx variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type HMI is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the `DigAct` block.



SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Display Format	`Stx.TPosFb`	#####	**Configuration**	1	Sets display format
Position fb delay (sec)	`Stx.TPosFb`	0	**Configuration**	2	Sets position feedback delay
Type Selection	`Typ`	0 - monostable	**Configuration**	7	Type selection
Default Position	`DefaultPos`	0 - close	**Configuration**	9	Default position

Name	Variable	Initial Value	Property Tab	Position	Description
Area Information	`AreaInfo`	Not available	**Device Data**	1	Area information
Asset ID	`AssetID`	Not available	**Device Data**	7	Asset ID

DigAct - Representation in the HMI

Digital Actuator (Genie)

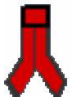
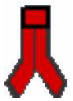
The DigAct block is represented in the HMI by the following genie.

Genie Elements

Symbol	Meaning
	Opens the DigAct popup.
	The instance name of the DigAct block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.
	Genie for DigAct as damper.
	Genie for DigAct as valve.

Animation Description of the Genie

Symbol	Animation	Meaning
	Line color: Light gray Fill color: Light gray Top color: Light gray	Digital actuator is in Auto mode and sleeping.
	Line color: Dark gray	Digital actuator is awake or in Manual mode.

Symbol	Animation	Meaning
	Line color: Dark gray Left fill color: White Right fill color: Flashing green Top color: Flashing green	Digital actuator has been started, valve or gate close command is given, and feedback yet to be received.
	Line color: Dark gray Left fill color: White Right fill color: Steady green Top color: Steady green	Digital actuator is running, and valve or gate is closed.
	Line color: Dark gray Left fill color: Light gray Right fill color: Steady green Top color: Light gray	Digital actuator is not ready, and valve or gate is closed.
	Line color: Dark gray Left fill color: White Right fill color: Steady green Top color: White	Digital actuator is ready, and valve or gate is closed.
	Line color: Dark gray Left fill color: Flashing green Right fill color: White Top color: Flashing green	Digital actuator has been started, valve or gate open command is given, and feedback yet to be received.
	Line color: Dark gray Left fill color: Steady green Right fill color: White Top color: Steady green	Digital actuator is running, and valve or gate is opened.
	Line color: Dark gray Left fill color: Steady green Right fill color: Light gray Top color: Light gray	Digital actuator is not ready, and valve or gate is opened.
	Line color: Dark gray Left fill color: Steady green Right fill color: White Top color: White	Digital actuator is ready, and valve or gate is opened.
	Line color: Dark gray Left fill color: Flashing red Right fill color: Flashing red Top color: Flashing red	Digital actuator is tripped, and a trip alarm is active.
	Line color: Dark gray Left fill color: Steady red Right fill color: Steady red Top color: Steady red	Digital actuator is tripped, and a trip alarm is acknowledged.

Modes

On the top left corner of the digital actuator genie, the mode of operation is indicated.

Mode	Meaning
A	Digital actuator is operating in the Auto mode.
M	Digital actuator is operating in the Manual mode.
O	Digital actuator is operating in the Out of Service mode.

Item	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Alarm Masking

The following table describes the animations for the genie when the alarms are masked:

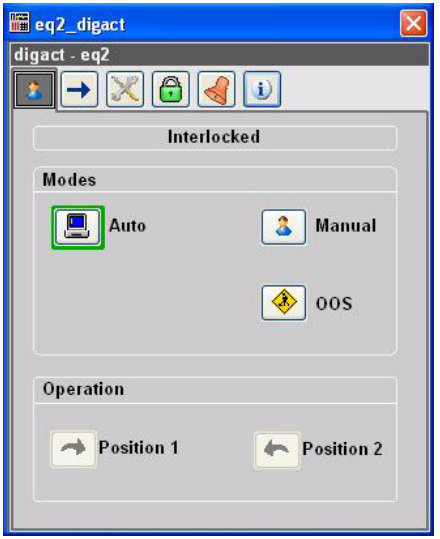
Symbol	Meaning
	The trip or alert alarms in the digital actuator are masked.

Popup

The `DigAct` block popup has the following tabs:

- **Operator** tab
- **Extended** tab
- **Configuration** tab
- **Interlock** tab
- **Alarm** tab
- **Information** tab

Operator Tab



This tab has 3 sub-sections:

- **Status**
- **Modes**
- **Operation**

The sub-section **Status** displays the current status, and the possible displays are listed as follows:

- **Sleeping**
- **Group Active**
- **Ready**
- **Not Ready**
- **Position 1 Reached**
- **Position 2 Reached**
- **Running**
- **Interlocked**
- **Warning**
- **Trip**

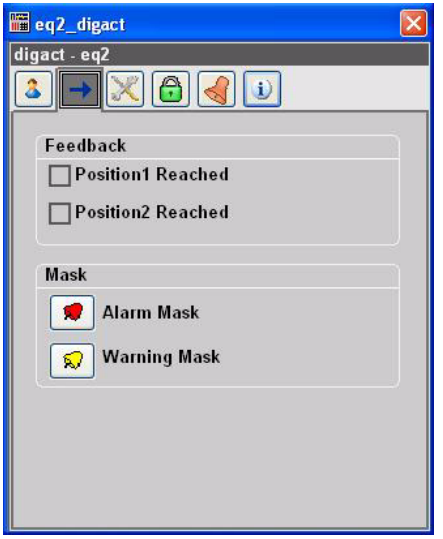
The following table describes the sub-section **Modes**:

Item	Meaning
Auto	The motor runs in Auto mode controlled by PLC program.
Manual	The motor runs in Manual mode controlled by HMI.
OOS	The motor runs in Out of Service mode for maintenance purpose of the object (controlled in Manual and Auto modes).

The following table describes the sub-section **Operation**:

Item	Meaning
Position 1	**Start** command is given to actuator to reach position 1.
Position 2	**Start** command is given to actuator to reach position 2.

Extended Tab



This tab has 2 sub-sections:

- **Feedback**
- **Mask**

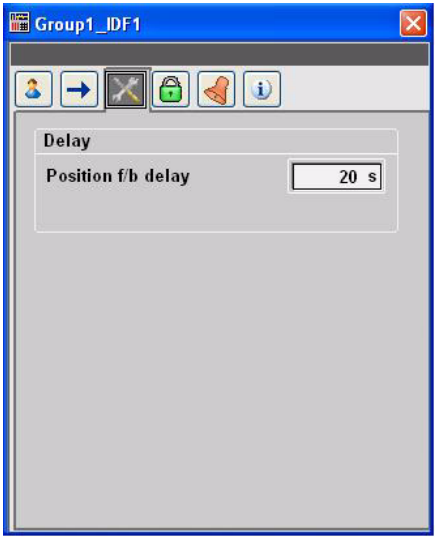
The following table describes the sub-section **Feedback**:

Item	Animation	Description
Position 1 Reached	Steady green	Feedback is obtained.
	Steady gray	Feedback is not obtained.
Position 2 Reached	Steady green	Feedback is obtained.
	Steady gray	Feedback is not obtained.

The following table describes the sub-section **Mask**:

Item	Description
Alarm Mask	Masks the indicated current trip alarms of this block in the alarm list.
Warning Mask	Masks the current alerts of this block in the alarm list.

Configuration Tab



This tab has a sub-section called as **Delay**, which is described in the following table:

Item	Description
Position f/b delay	Displays the position feedback delay of motor, configurable from HMI. After starting the motor, if position feedback is not received before delay time, an alarm is generated, and actuator is tripped.

Interlock Tab

This tab has a sub-section called as **Interlock**, which is described in the following table:

Item	Animation	Description
Starting Interlock	Steady green	**Starting Interlock** is active, and the related device is not blocked.
	Steady gray	**Starting Interlock** is not active, and the related device is blocked.
Safety Interlock	Steady green	**Safety Interlock** is active, and the related device is not blocked.
	Steady gray	**Safety Interlock** is not active, and the related device is blocked.
Sequential Interlock	Steady green	**Sequential Interlock** is active, and the related device is not blocked.
	Steady gray	**Sequential Interlock** is not active, and the related device is blocked.
Process Interlock	Steady green	**Process Interlock** is active, and the related device is not blocked.
	Steady gray	**Process Interlock** is not active, and the related device is blocked.

This tab has sub-tabs Interlock, **STI**, **SI**, and **SQI**.

For description about **Interlock** sub-tabs, refer to Commonly Used Tabs *(see page 53)*.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

DigAct_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type DigAct_St for the DigAct block:

Element	Type	Description
TPosFb	INT	Position feedback delay
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Stw2	WORD	Status word 2 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI
Dummy	WORD	Not used

Bit description for the Ctw element:

Bit	Description
0	Auto mode selection from HMI
1	Manual mode selection from HMI
2	Not used
3	Out of Service mode selection from HMI
4	Close
5	Open
6	Stop
7	Alarm reset from HMI
8...13	Not used
14	Alert mask
15	Alarm mask

Bit description for the Stw1 element:

Bit	Description
0	Auto mode selection from HMI
1	Not used
2	Group wake
3	Type selection
4	Start interlock
5	Safety interlock

Bit	Description
6	Sequential interlock
7	Process interlock
8...12	Not used
13	Position 1 feedback
14	Alarm inhibition
15	Position 2 feedback

Bit description for the `Stw2` element:

Bit	Description
0	Auto mode active status
1	Manual mode active status
2	Not used
3	Out of Service mode active status
4	Output `OP1` active
5	Output `OP2` active
6	Motor ready status
7	Run
8...9	Not used
10	Position 1 reached
11	Position 2 reached
12	Interlock active status
13	Group active status
14	Alert active status
15	Trip active status

Bit description for the `Alw1` element:

Bit	Description
0	Not used
1	Safety interlock not available
2	Sequential interlock not available
3	Proces interlock not available
4...10	Not used
11	Position 1 feedback is not active
12	Position 2 feedback is not active

Bit	Description
13	Position1 and position 2 feedback are active
14	Not used
15	Active trip alarm

8.9 MotAct: Motor Actuator

Overview

This section describes the MotAct block.

What's in this Section?

This section contains the following topics:

Topic	Page
MotAct - Description	261
MotAct - Representation as SCoD in UAG	272
MotAct - Representation in the HMI	275
MotAct_St Derived Data Type	284

MotAct - Description

General

The MotAct block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB MotAct
Vijeo Citect	Genie MotAct
UAG	SCoD MotAct

Function Description

The MotAct block is used to control electrical actuators with defined positions used for driving valves, gates, dampers, and so on.

This block supports 3 modes of operation:
- Auto mode
- Manual mode
- Local mode

By default, the motor will be in Auto mode. The motor is forced to Auto or Local mode if the mode selection input pin (AutoSel or LOC) is HIGH. The Local pin always has highest priority.

When the Local mode is selected, the block functions in Local mode irrespective of other mode selections.

If the mode selection input pin (AutoSel or LOC) is LOW, operator changes mode of the motor from the popup of motor block. When the motor is under maintenance, the Out of Service mode can be activated from the popup.

Auto Mode Operation

A wake-up command (**Wake**) is given by the GrpCtrl block to wake-up the motor from sleep. A start-up pulse command to achieve position is given either from the GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group.

If Direction is 0, this sets the output OP1 as HIGH and actuator starts to selected position1.

If Direction is 1, this sets the output OP2 as HIGH and actuator starts to selected position2.

After the start command (**OP1**) or (**OP2**) is issued to the actuator within the specified position feedback delay, the position feedback has to be obtained.

If the position feedback is not obtained, the actuator is tripped, and an alarm is generated. If the position feedback is obtained, the actuator stops.

Direction change of the actuator demands stoppage of the actuator and a time delay before restarting the actuator in other direction.

`Defaultpos` is stop position which defines the position at which actuator must stop when **Stop** command is received. If `DefaultPos` is 1, stop position is position1and if `DefaultPos` is 0, stop position is position 2.

The block sets an output group active (`GActive`) if the motor is started in Auto mode. If the block is group active and actuator has reached desired position, a pulse command (**SeqStart**) is given to the next motor to start.

Group stop command (**Stop**) is received by the actuator from the `GrpCtrl` or `GrpStpMgt` or `RouteMgt` or previous device in the group.

Group stop command (**Stop**) or group quick stop command (**QStop**) sets the actuator to stop position, that is, it starts the actuator in the stop position (`DefaultPos`) direction, and then stops.

Group quick stop command (**QStop**) is received by the motor actuator from the `GrpCtrl` block. If the block is group shut down, an output command (**SeqStop**) is given to the next motor actuator to stop.

Manual Mode Operation

The operator initiates the **Position1** and **Position2** commands from the popup of the motor actuator block. Direction change, run, and position feedback monitoring are applicable as in Auto mode.

Local Mode Operation

Start-up to position 1 pulse command (**LStart1**) or start-up to position 2 pulse command (**LStart2**) comes from the local control panel of the motor that sets the output `OP1` or `OP2` HIGH and actuator starts to selected position.

Run, and position feedback monitoring are applicable as in Auto mode.

Interlock Types

There are 4 types of interlocks. The result of individual interlock condition blocks is connected to these interlock pins:

Interlock Name	Parameter	Description
Starting Interlock	`StartILck`	These interlocks are valid before motor start-up only.
Safety Interlock	`SftyILck`	All the safety conditions are taken into consideration. This input is active in all the modes.

Interlock Name	Parameter	Description
Sequential Interlock	SeqILck	All the sequential conditions are taken into consideration. This input is active only in Auto mode.
Process Interlock	ProcILck	All the process conditions are taken into consideration. This input is active in Auto and Manual mode.

Feedback

The status of the interlock is displayed in the popup. Detailed description of the interlocks is displayed in the **Interlock** tab of the popup.

The availability (Avail) of the motor is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated. The overload feedback (OverLoad) of the motor is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated.

An emergency stop signal (EStop) comes from MCC panel emergency shut-down push button and is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated.

The alert and trip are indicated in the SCADA. The number of active alarms is shown in the popup of the motor. The details of the active alarms are shown in the **Alarm** tab of the popup. Based on configured conditions, the alarms for this block can be inhibited by connecting the logic to the input pin AlmInh. The input pin AlmRst that is connected from GrpCtrl block or the operator from popup can reset the trip alarms.

Configuration

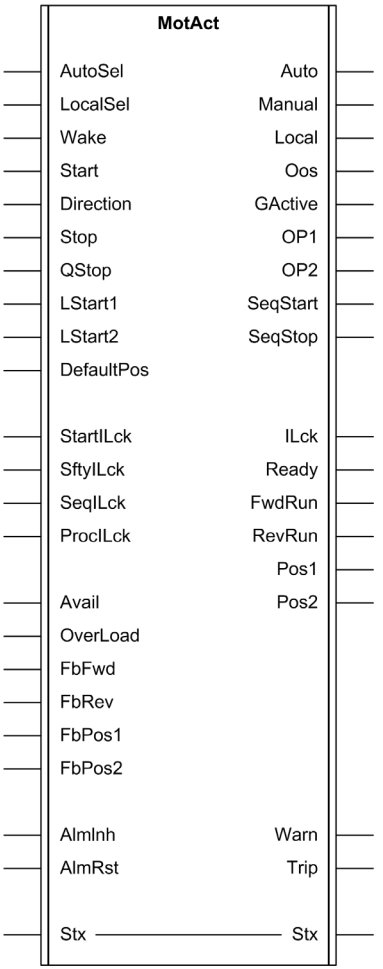
The run hours of the motor is displayed in the **Configuration** tab of the popup. After the maintenance of the motor, the run hours can be reset from the popup.

NOTE: The following parameters in the **Configuration** tab have user restriction from the HMI:

- **Run feedback delay**
- **Position feedback delay**
- **Inverse direction start delay**

Representation in FBD

The following figure shows the FBD representation of the `MotAct` block.



Parameter Description

The following table describes the input parameters of the MotAct block:

Parameter	Type	Description	Details
AutoSel	BOOL	Auto mode selection	Selection of operation mode (Auto mode): • AutoSel = 1: Block is forced to Auto mode. Mode selection option in HMI is grayed out. • AutoSel = 0: Block is set free for operator to select the mode from HMI.
LocalSel	BOOL	Local mode selection	Selection of operation mode (Local mode): • LocalSel = 1: Block is forced to Local mode. Mode selection option in HMI is grayed out. • LocalSel = 0: Block is set free for operator to select the mode from HMI.
Wake	BOOL	Group wake-up	Group wake-up and request for group availability check: • Wake = 1: Group is active. • Wake = 0: Group is in Sleep mode. This input is connected from output GWake of GrpCtrl.
Start	BOOL	Group start	**Start** command to the motor. GStart1 = 1: OP pin is set to 1, this is applicable in Auto mode only. This input is connected from the GStart pin of GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group.
Direction	BOOL	Direction	• Direction = 1: Actuator position is opened (Pos2). • Direction = 0: Actuator position is closed (Pos1).
Stop	BOOL	Group stop	**Stop** command to the motor. GStop = 1: OP pin is set to 0, this is applicable in Auto mode only. This input is connected from the GStop pin of GrpCtrl or GrpStpMgt or RouteMgt or previous device in the group.

Parameter	Type	Description	Details
QStop	BOOL	Group quick stop	**Stop** command to the motor. QStop = 1: OP pin is set to 0, this is applicable in Auto mode only. This input is connected from the GQStop pin of GrpCtrl.
LStart1	BOOL	Local start in forward direction	**Start** command in forward direction to the motor. LStart1 = 1: OP pin is set to 1, this is applicable in Local mode only. This input is connected from the **Start** push button of Local IO panel.
LStart2	BOOL	Local start in backward direction	**Start** command in backward direction to the motor. LStart2 = 0...1: OP pin is set to 1, this is applicable in Local mode only. This input is connected from the **Start** push button of Local IO panel.
DefaultPos	BOOL	Default position	• DefaultPos = 1: Actuator position is opened (Pos2). • DefaultPos = 0: Actuator position is closed (Pos1).
StartILck	BOOL	Starting interlock	• StartILck = 1: The starting interlock conditions are operational. • StartILck = 0: The starting interlock conditions are not operational. Applicable only in Auto and Manual modes of operation. Applicable only before starting of the motor. After motor is started, StartILck is not considered.
SftyILck	BOOL	Safety interlock	• SftyILck = 1: All the safety interlock conditions are operational. • SftyILck = 0: All the safety interlock conditions are not operational, trips the motor and generates an alarm. Applicable in all the modes of operation (Auto, Manual, and Local).

Parameter	Type	Description	Details
SeqILck	BOOL	Sequential interlock	• SeqILck = 1: All the sequential interlock conditions are operational. • SeqILck = 0: All the sequential interlock conditions are not operational, trips the motor and generates an alarm. Applicable in Auto mode only.
ProcILck	BOOL	Process interlock	• ProcILck = 1: All the process interlock conditions are operational. • ProcILck = 0: All the process interlock conditions are not operational, trips the motor and generates an alarm. Applicable in Auto and Manual modes only.
Avail	BOOL	Availability of motor	• Avail = 1: Motor is electrically available. • Avail = 0: Motor is electrically not available. **Stop** command to the motor, OP = 0. This signal comes from the MCC outgoing circuit (electrical protection active, circuit breaker switched off, drawer out, and so on).
OverLoad	BOOL	Overload	• OverLoad = 1: Motor is in overload condition. • OverLoad = 0: Motor is not in overload condition. **Stop** command to the motor, OP = 0. The signal comes from the MCC outgoing circuit (electrical protection active).
FbFwd	BOOL	Forward run feedback of motor - electrical	Once the motor is given a **Start** command in forward direction from output OP1, its electrical run feedback is continuously monitored. This signal comes from the MCC outgoing circuit. If OP1 = 1 and FbFwd = 0, an alarm is generated after time delay. If OP1 = 0 and FbFwd = 1, an alarm is generated after time delay.

Parameter	Type	Description	Details
FbRev	BOOL	Reverse run feedback of motor - electrical	Once the motor is given a **Start** command in reverse direction from output OP2, its electrical run feedback is continuously monitored. This signal comes from the MCC outgoing circuit. If OP2 = 1 and FbRev = 0, an alarm is generated after time delay. If OP2 = 0 and FbRev = 1, an alarm is generated after time delay.
FbPos1	BOOL	Position feedback1	Once the actuator is given a **Start** command in forward direction from output OP1, its position feedback is continuously monitored. If Pos1 = 1: **Stop** command to the actuator, OP1 = 0 is given. If OP1 = 1 and FbPos1 = 0 for more than time delay TFbPos, an alarm is generated.
FbPos2	BOOL	Position feedback2	Once the actuator is given a **Start** command in forward direction from output OP2, its position feedback is continuously monitored. If Pos2 = 1: **Stop** command to the actuator, OP2 = 0 is given. If OP2 = 1 and FbPos2 = 0 for more than time delay TFbPos, an alarm is generated.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

The AlmInh parameter to inhibit alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the MotAct block:

Parameter	Type	Description	Details
AlmInh	BOOL	Alarm inhibition	• AlmInh = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • AlmInh = 0: Alarms are not inhibited.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

To reset, alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the MotAct block:

Parameter	Type	Description	Details
AlmRst	BOOL	Reset	• AlmRst = 1: Resets all the alarms generated by the block. • AlmRst = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from GRst pin of GrpCtrl block and resets the trip alarm.

The following table describes the output parameters of the MotAct block:

Parameter	Type	Description	Details
Auto	BOOL	Auto mode	Auto = 1: Auto mode is selected. Motor operation is igroup based on GStart and GStop.
Manual	BOOL	Manual mode	Manual = 1: Manual mode is selected. Motor operation is based on commands **Start** and **Stop** given by operator from HMI.
Local	BOOL	LocalManual mode	Local = 1: Local mode is selected. Motor operation is based on commands LStart and LStop given from field.

Parameter	Type	Description	Details
Oos	BOOL	Out of Service mode	Oos = 1: Out of Service mode is selected. Motor is not operated and OP = 0.
GActive	BOOL	Group active	The motor is started in the group and is active in the group. Start command pulse **Start** is obtained, GActive = 1.
OP1	BOOL	Output command to run motor in forward direction	OP1 = 1: Motor is given command to close.
OP2	BOOL	Output command to run motor in reverse direction	OP2 = 1: Motor is given command to open.
SeqStart	BOOL	**Start** command to next motor in group	Start command (pulse) is received and outputs OP and Run = 1, sequential start command pulse is given to the next motor connected in the group.
SeqStop	BOOL	**Stop** command to next motor in group	Stop or QStop = 1 and OP and Run = 0, sequential stop command is given to the next motor connected in the group.
ILck	BOOL	Interlock status	• ILck = 1: All the interlocks are operational. • ILck = 0: All the interlocks are not operational. There is an active interlock present.
Ready	BOOL	Motor ready to start	• Ready = 1: All the conditions to start the motor are operational. • Ready = 0: The conditions to start the motor are not operational. Motor will not be given **Start** command.
FwdRun	BOOL	Forward run	Motor is running in forward direction.
RevRun	BOOL	Reverse run	Motor is running in reverse direction.
Pos1	BOOL	Actuator in position1	Actuator has reached position1. Applicable in all the modes of operation.
Pos2	BOOL	Actuator in position2	Actuator has reached position2. Applicable in all the modes of operation.

Parameter	Type	Description	Details
Warn	BOOL	Alert	Warn = 1: The motor is operational with no active alert present. Warn = 0: The motor is operational with an active alert present.
Trip	BOOL	Trip	Trip = 1: The motor is operational with no active trip alarm present. Trip = 0: The motor has tripped with an active trip alarm present.

The following table describes the input/output parameter of the MotAct block:

Parameter	Type	Description
Stx	MotAct_St	Status and control of the motor actuator. Refer to the Structure to interface with the HMI *(see page 284)*.

MotAct - Representation as SCoD in UAG

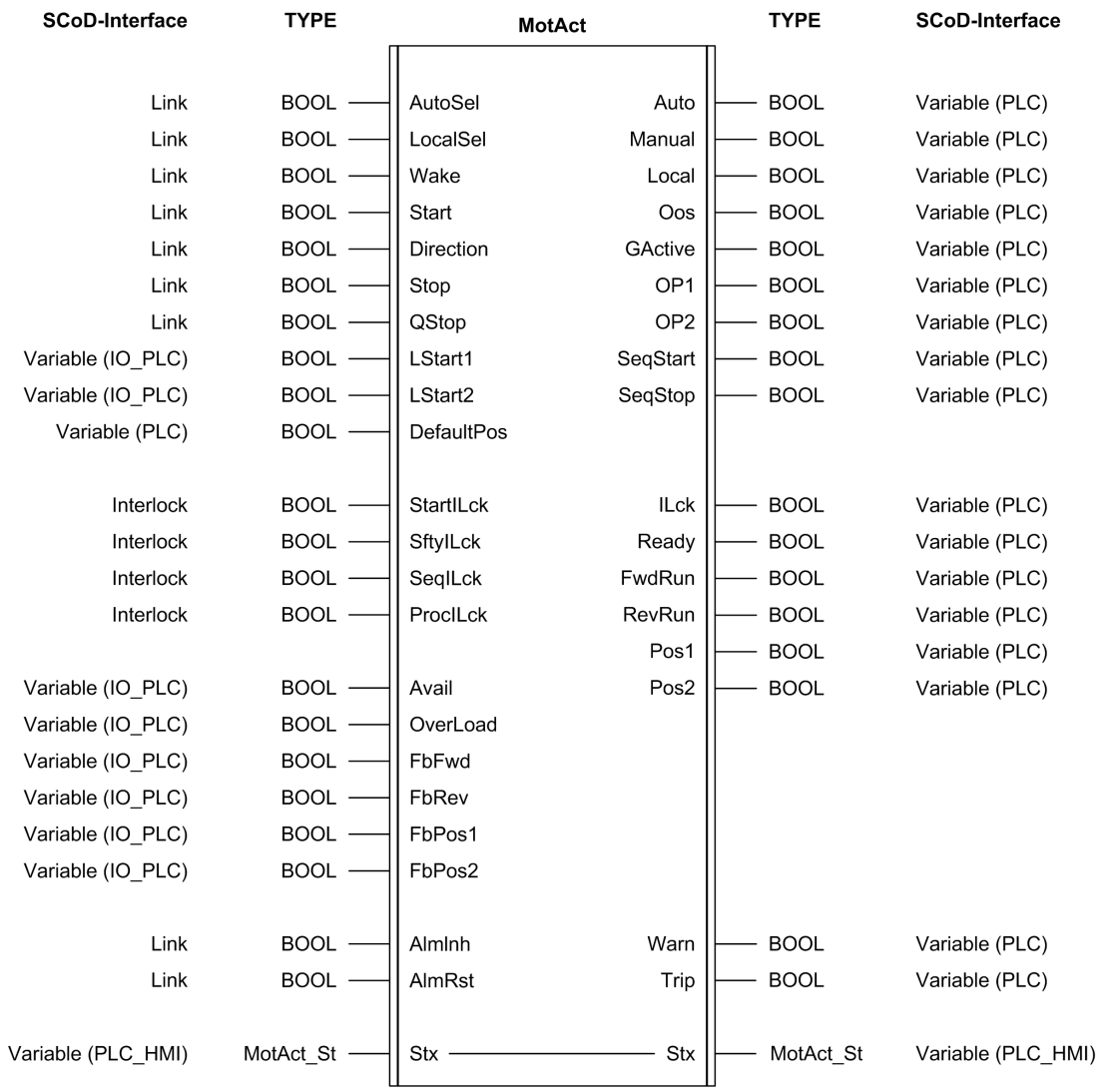
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the Stx variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type HMI is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the MotAct block.


SCoD-Interface
TYPE
MotAct
TYPE
SCoD-Interface
Link
BOOL
AutoSel
Auto
BOOL
Variable (PLC)
Link
BOOL
LocalSel
Manual
BOOL
Variable (PLC)
Link
BOOL
Wake
Local
BOOL
Variable (PLC)
Link
BOOL
Start
Oos
BOOL
Variable (PLC)
Link
BOOL
Direction
GActive
BOOL
Variable (PLC)
Link
BOOL
Stop
OP1
BOOL
Variable (PLC)
Link
BOOL
QStop
OP2
BOOL
Variable (PLC)
Variable (IO_PLC)
BOOL
LStart1
SeqStart
BOOL
Variable (PLC)
Variable (IO_PLC)
BOOL
LStart2
SeqStop
BOOL
Variable (PLC)
Variable (PLC)
BOOL
DefaultPos
Interlock
BOOL
StartILck
ILck
BOOL
Variable (PLC)
Interlock
BOOL
SftyILck
Ready
BOOL
Variable (PLC)
Interlock
BOOL
SeqILck
FwdRun
BOOL
Variable (PLC)
Interlock
BOOL
ProcILck
RevRun
BOOL
Variable (PLC)
Pos1
BOOL
Variable (PLC)
Variable (IO_PLC)
BOOL
Avail
Pos2
BOOL
Variable (PLC)
Variable (IO_PLC)
BOOL
OverLoad
Variable (IO_PLC)
BOOL
FbFwd
Variable (IO_PLC)
BOOL
FbRev
Variable (IO_PLC)
BOOL
FbPos1
Variable (IO_PLC)
BOOL
FbPos2
Link
BOOL
AlmInh
Warn
BOOL
Variable (PLC)
Link
BOOL
AlmRst
Trip
BOOL
Variable (PLC)
Variable (PLC_HMI)
MotAct_St
Stx
Stx
MotAct_St
Variable (PLC_HMI)

SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Display Format (time)	`Stx.RunHr`	#####	**Configuration**	1	Display format
Run fb delay (sec)	`Stx.TRunFb`	0	**Configuration**	2	Sets run feedback delay (s)
Inv direction start delay (sec)	`Stx.TInv`	0	**Configuration**	3	Sets inversion direction start delay (s)
Default Position	`DefaultPos`	0 - close	**Configuration**	7	Default position
Position fb delay (sec)	`Stx.TPosFb`	0	**Configuration**	8	Sets position feedback delay (s)
Area Information	`AreaInfo`	Not available	**Device Data**	1	Area information
Asset ID	`AssetID`	Not available	**Device Data**	7	Asset ID

MotAct - Representation in the HMI

Motor Actuator (Genie)

The MotAct block is represented in the HMI by the following genie.

Genie Elements

Symbol	Meaning
	Opens the MotAct popup.
	The instance name of the MotAct block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.
	Genie for MotAct as damper.
	Genie for MotAct as valve.

Animation Description of the Genie

Symbol	Animation	Meaning
	Line color: Light gray Fill color: Light gray Top color: Light gray	Motor actuator is in Auto mode and sleeping.
	Line color: Dark gray	Motor actuator is awake or in Local or Manual mode.
	Line color: Dark gray Left fill color: White Right fill color: Flashing green Top color: Flashing green	Motor actuator has been started, and valve or gate close command is given, feedback yet to be received.

Symbol	Animation	Meaning
	Line color: Dark gray Left fill color: White Right fill color: Steady green Top color: Steady green	Motor actuator is running, and valve or gate is closed.
	Line color: Dark gray Left fill color: Light gray Right fill color: Steady green Top color: Light gray	Motor actuator is not ready, and valve or gate is closed.
	Line color: Dark gray Left fill color: White Right fill color: Steady green Top color: White	Motor actuator is ready, and valve or gate is closed.
	Line color: Dark gray Left fill color: Flashing green Right fill color: White Top color: Flashing green	Motor actuator has been started, and valve or gate open command is given, feedback yet to be received.
	Line color: Dark gray Left fill color: Steady green Right fill color: White Top color: Steady green	Motor actuator is running, and valve or gate is opened.
	Line color: Dark gray Left fill color: Steady green Right fill color: Light gray Top color: Light gray	Motor actuator is not ready, and valve or gate is opened.
	Line color: Dark gray Left fill color: Steady green Right fill color: White Top color: White	Motor actuator is ready, and valve or gate is opened.
	Line color: Flashing yellow	Motor actuator is running, valve or gate is in closed position, and an alert alarm is active.
	Line color: Steady yellow	Motor actuator is running, valve or gate is in closed position, and an alert alarm is acknowledged.
	Line color: Dark gray Fill color: Flashing red Top color: Flashing red	Motor actuator is tripped, and a trip alarm is active.
	Line color: Dark gray Fill color: Steady red Top color: Steady red	Motor actuator is tripped, and a trip alarm is acknowledged.

Modes

On the top left corner of the motor actuator genie, the mode of operation is indicated.

Mode	Meaning
A	Motor actuator is operating in the Auto mode.
M	Motor actuator is operating in the Manual mode.
L	Motor actuator is operating in the Local mode.
O	Motor actuator is operating in the Out of Service mode.

Item	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Alarm Masking

The following table describes the animations for the genie when the alarms are masked:

Symbol	Meaning
	The trip or alert alarms in the motor actuator are masked.

Popup

The `MotAct` block popup has the following tabs:

- **Operator** tab
- **Extended** tab
- **Configuration** tab
- **Interlock** tab
- **Alarm** tab
- **Information** tab

Operator Tab

This tab has 3 sub-sections:

- **Status**
- **Modes**
- **Operation**

The sub-section **Status** displays the current status, and the possible displays are listed as follows:

- **Sleeping**
- **Group Active**
- **Ready**
- **Not Ready**
- **Position 1 Reached**
- **Position 2 Reached**
- **Running**
- **Interlocked**
- **Warning**
- **Trip**

The following table describes the sub-section **Modes**:

Item	Meaning
Auto	The motor runs in Auto mode controlled by PLC program.
Manual	The motor runs in Manual mode controlled by HMI.

Item	Meaning
Local	The motor runs in Local mode controlled by local control box of the motor.
OOS	The motor runs in Out of Service mode for maintenance purpose of the object (applicable in all modes).

The following table describes the sub-section **Operation**:

Item	Meaning
Position 1	**Start** command is given to actuator to reach position 1.
Position 2	**Start** command is given to actuator to reach position 2.

Extended Tab

This tab has 2 sub-sections:

- **Feedback**
- **Mask**

The following table describes the sub-section **Feedback**:

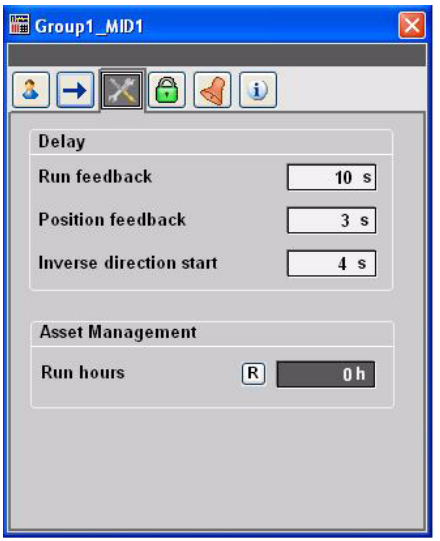
Item	Animation	Description
Available	Steady green	Motor electrically available feedback is obtained.
	Steady gray	Motor electrically not available feedback is obtained.

Item	Animation	Description
Over Load	Steady green	Motor overload feedback is obtained.
	Steady gray	Motor overload feedback is not obtained.
Forward direction	Steady green	Motor is active and running in forward direction.
	Steady gray	Motor is not active.
Reverse direction	Steady green	Motor is active and running in reverse direction.
	Steady gray	Motor is not active.
Position1 Reached	Steady green	Position 1 reached, feedback is obtained.
	Steady gray	Position 1 reached, feedback is not obtained.
Position2 Reached	Steady green	Position 2 reached, feedback is obtained.
	Steady gray	Position 2 reached, feedback is not obtained.

The following table describes the sub-section **Mask**:

Item	Description
Alarm Mask	Masks the indicated current trip alarms of this block in the alarm list.
Warning Mask	Masks the current alerts of this block in the alarm list.

Configuration Tab



This tab has 2 sub-sections:

- **Delay**
- **Asset Management**

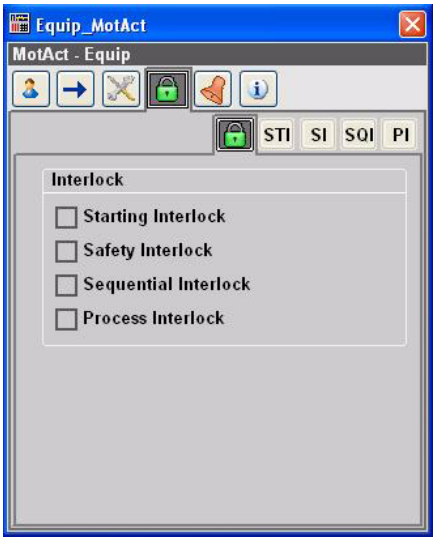
The following table shows the sub-section **Delay**:

Item	Description
Run Feedback	Displays run feedback delay of motor, configurable from HMI. After starting the motor if run feedback is not received before delay time, an alarm is generated, and motor is tripped.
Position feedback	Displays position feedback delay of motor, configurable from HMI. After starting the motor, if position feedback is not received before delay time, an alarm is generated, and motor is tripped.
Inverse direction start	Inverse direction starts delay of motor, configurable from HMI. Time taken to switch on motor from forward to reverse direction or vice versa exceeds delay time, an alarm is generated, and motor is tripped.

The following table shows the sub-section **Asset Management**:

Item	Meaning
Run hours	Displays the run hours of motor.
	Resets the run hours value of the motor to 0.

Interlock Tab



This tab has a sub-section called as **Interlock**, which is described in the following table:

Item	Animation	Description
Starting Interlock	Steady green	**Starting Interlock** is active, and the related device is not blocked.
	Steady gray	**Starting Interlock** is not active, and the related device is blocked.
Safety Interlock	Steady green	**Safety Interlock** is active, and the related device is not blocked.
	Steady gray	**Safety Interlock** is not active, and the related device is blocked.
Sequential Interlock	Steady green	**Sequential Interlock** is active, and the related device is not blocked.
	Steady gray	**Sequential Interlock** is not active, and the related device is blocked.
Process Interlock	Steady green	**Process Interlock** is active, and the related device is not blocked.
	Steady gray	**Process Interlock** is not active, and the related device is blocked.

This tab has sub-tabs Interlock, **STI**, **SI**, and **SQI**.

For description about **Interlock** sub-tabs, refer to Commonly Used Tabs *(see page 53)*.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

MotAct_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type MotAct_St for the MotAct block:

Element	Type	Description
RunHr	INT	Run hours of motor
TRunFb	INT	Run feedback delay
TPosFb	INT	Position feedback delay
TInv	INT	Inverse direction delay
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Stw2	WORD	Status word 2 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI

Bit description for the Ctw element:

Bit	Description
0	Auto mode selection from HMI
1	Manual mode selection from HMI
2	Local mode selection from HMI
3	Out of Service mode selection from HMI
4	Start 1 command from HMI
5	Start 2 command from HMI
6	Stop command from HMI
7	Alarm reset from HMI
8	Run hour reset from HMI
9...13	Not used
14	Alert mask
15	Alarm mask

Bit description for the Stw1 element:

Bit	Description
0	Auto mode selection from HMI
1	Local mode selection from HMI
2	Group wake

Bit	Description
3	Not used
4	Starting interlock
5	Safety interlock
6	Sequential interlock
7	Process interlock
8	Motor electrically available
9	Motor overload
10	Not used
11	Run feedback forward direction
12	Run feedback reverse direction
13	Position 1 feedback
14	Alarm inhibition
15	Position 2 feedback

Bit description for the `Stw2` element:

Bit	Description
0	Auto mode active status
1	Manual mode active status
2	Local mode active status
3	Out of Service mode active status
4	Output command OP1 active
5	Output command OP2 active
6	Motor ready status
7	Run
8	Forward run
9	Reverse run
10	Position 1 reached
11	Position 2 reached
12	Interlock active status
13	Group active status
14	Alert active status
15	Trip active status

Bit description for the `Alw1` element:

Bit	Description
0	Not used
1	Safety interlock not available
2	Sequential interlock not available
3	Process interlock not available
4	Motor electrically not available
5	Motor overload
6	Not used
7	Run feedback not active when motor is running
8	Run feedback active when motor is not running
9	Reverse feedback is not active when motor is running
10	Reverse feedback is active when motor is not running
11	Position 1 feedback is not active
12	Position 2 feedback is not active
13	Position1 and position 2 feedback is high
14	Run hours maximum value reached
15	Active trip alarm

8.10 AnlgAct: Analog Actuator

Overview

This section describes the AnlgAct block.

What's in this Section?

This section contains the following topics:

Topic	Page
AnlgAct - Description	288
AnlgAct - Representation as SCoD in UAG	300
AnlgAct - Representation in the HMI	303
AnlgAct_St Derived Data Type	313

AnlgAct - Description

General

The AnlgAct block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB AnlgAct
Vijeo Citect	Genie AnlgAct
UAG	SCoD AnlgAct

Function Description

The AnlgAct block is used to control the position of equipment by a 2 way driven actuator that can be energised by an electric motor, an electrical jack, and hydraulic jack with solenoid valves.

The function of the block deals only with correct positioning of the actuator itself for a given set value rather than any kind of process related regulation. In case process related positioning is required, the block must be used together with a PIDB block. This block is used for controlling analog actuators with 2 directions of rotation.

This block supports 3 modes of operation:
- Auto mode
- Manual mode
- Local mode

By default, the motor will be in Auto mode. The analog actuator is forced to Auto or Local mode if the mode selection input pin (AutoSel or LOC) is HIGH. The Local pin always has highest priority.

When the Local mode is selected, the block functions in Local mode irrespective of other mode selections.

If the mode selection input pin (AutoSel or LOC) is LOW, operator changes mode of the analog actuator from the popup of analog actuator block. When the analog actuator is under maintenance, the Out of Service mode can be activated from the popup.

Auto Mode Operation

A wake-up command (**Wake**) is given by the GrpCtrl block to wake-up the actuator from sleep. A start-up pulse command (**Start**) is given either from the GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group. This sets the output OP1 or OP2 as HIGH depending on deviation (that is, if deviation is less than SP start OP2 and if deviation is more than SP start OP1).

For device safety, OP1 is not set if FbPos1 is HIGH and OP2 is not set if FbPos2 is HIGH.

After the start command **OP1** or **OP2** is issued to the actuator, the forward or reverse run feedback has to be obtained.

If the run feedback is not obtained within the specified run feedback delay, the actuator is tripped and an alarm is generated.

Direction change of the actuator demands stoppage of the actuator and a time delay before restarting the actuator in other direction.

`DefaultPos` is stop position which defines the position at which actuator must stop when **Stop** command is received.

The block sets an output group active (`GActive`) if the actuator is started in Auto mode. If the block is group active and actuator has reached set point, a pulse command (**SeqStart**) is given to the next actuator to start.

Group stop command (**Stop**) and group quick stop command (**QStop**) are received by the actuator from the `GrpCtrl` or `GrpStpMgt` or `RoutMgt` or previous device in the group.

Group stop command (**Stop**) or group quick stop command (**QStop**) sets the actuator to stop position (that is, it changes the set point to stop position (`DefaultPos`) and actuator stops once the `PV` reaches the set point).

If the block is group shut down, an output command (**SeqStop**) is given to the next actuator to stop.

Manual Mode Operation

The operator initiates the **Run** and **Stop** commands from the popup of the analog actuator block. Run and position feedback monitoring are applicable as in Auto mode. The operator enters the position set point from popup.

Local Mode Operation

Forward and reverse start commands (**LStart1** and **LStart2**) are given from field by push button. These commands inturn sets `OP1` and `OP2` till the continuous input is received.

Safety consideration of `FbPos1` and `FbPos2`, run and process feedback monitoring are applicable as in Auto mode.

Interlock Types

There are 4 types of interlocks. The result of individual interlock condition blocks is connected to these interlock pins:

Interlock Name	Parameter	Description
Starting Interlock	StartILck	These interlocks are valid before motor start-up only.
Safety Interlock	SftyILck	All the safety conditions are taken into consideration. This input is active in all the modes.
Sequential Interlock	SeqILck	All the sequential conditions are taken into consideration. This input is active only in Auto mode.
Process Interlock	ProcILck	All the process conditions are taken into consideration. This input is active in Auto and Manual mode.

Feedback

The status of the interlock is displayed in the popup. Detailed description of the interlocks is displayed in the **Interlock** tab of the popup.

The availability (Avail) of the motor is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated. The overload feedback (OverLoad) of the motor is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated.

The alert and trip are indicated in the SCADA. The number of active alarms is shown in the popup of the motor. The details of the active alarms are shown in the **Alarm** tab of the popup. Based on configured conditions, the alarms for this block can be inhibited by connecting the logic to the input pin AlmInh. The input pin AlmRst that is connected from GrpCtrl block or the operator from popup can reset the trip alarms.

Configuration

You can configure the following parameters:

- Default value, that is, preferred position of the actuator which must be driven accordingly when it is not released anymore.
- Minimum deviation, that is, a pulse is given to OP1/OP2 only if |PV - SP| is greater than minimum deviation value.
- Pulse length time to tune precisely.
- Calculation time that is time interval between 2 calculation cycles.
- Movement time that is time required for the actuator to move from 1 end position to another.
- Minimum pulse time required to excite the driving motor or solenoid valve.

Maintenance

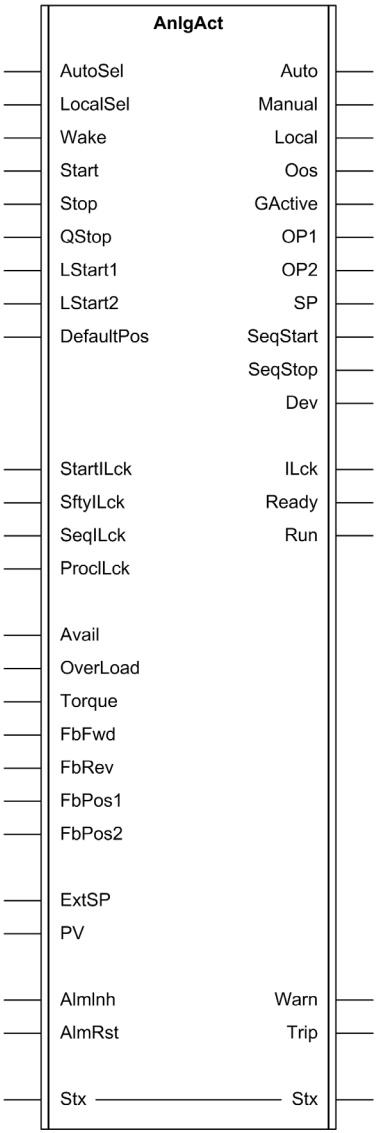
The run hours of the motor is displayed in the **Configuration** tab of the popup. After the maintenance of the motor, the run hours can be reset from the popup.

NOTE: The following parameters in the **Configuration** tab have user restriction from the HMI:

- **Run feedback delay**
- **Run hours reset**

Representation in FBD

The following figure shows the FBD representation of the `AnlgAct` block.



Parameter Description

The following table describes the input parameters of the AnlgAct block:

Parameter	Type	Description	Details
AutoSel	BOOL	Auto mode selection	Selection of operation mode (Auto mode): • AutoSel = 1: Block is forced to Auto mode. Mode selection option in HMI is grayed out. • AutoSel = 0: Block is set free for operator to select the mode from HMI.
LocalSel	BOOL	Local selection	Selection of operation mode (Local mode): • LocalSel = 1: Block is forced to Local mode. Mode selection option in HMI is grayed out. • LocalSel = 0: Block is set free for operator to select the mode from HMI.
Wake	BOOL	Group wake-up	Group wake-up and request for group availability check: • Wake = 1: Group is active. • Wake = 0: Group is in Sleep mode. This input is connected from output GWake of GrpCtrl.
Start	BOOL	Group start	**Start** command to the motor. GStart1 = 1: OP pin is set to 1, this is applicable in Auto mode only. This input is connected from the GStart pin of GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group.
Stop	BOOL	Group stop	**Stop** command to the motor. GStop = 1: OP pin is set to 0, this is applicable in Auto mode only. This input is connected from the GStop pin of GrpCtrl or GrpStpMgt or RouteMgt or previous device in the group.
QStop	BOOL	Group quick stop	**Stop** command to the motor. QStop = 1: OP pin is set to 0, this is applicable in Auto mode only. This input is connected from the GQStop pin of GrpCtrl.

Parameter	Type	Description	Details
LStart1	BOOL	Local start in forward direction	**Start** command in forward direction to the motor. LStart1 = 1: OP pin is set to 1, this is applicable in Local mode only. This input is connected from the **Start** push button of Local IO panel.
LStart2	BOOL	Local start in backward direction	**Start** command in backward direction to the motor. LStart2 = 1: OP pin is set to 1, this is applicable in Local mode only. This input is connected from the **Start** push button of Local IO panel.
DefaultPos	REAL	Default position	It determines the user preferred position of the actuator during stop.
StartILck	BOOL	Starting interlock	• StartILck = 1: All the starting interlock conditions are operational. • StartILck = 0: All the starting interlock conditions are not operational. Applicable only in Auto and Manual modes of operation. Applicable only before starting of the motor. After motor is started, StartILck is not considered.
SftyILck	BOOL	Safety interlock	• SftyILck = 1: All the safety interlock conditions are operational. • SftyILck = 0: All the safety interlock conditions are not operational, trips the motor and generates an alarm. Applicable in all the modes of operation (Auto, Manual, and Local).
SeqILck	BOOL	Sequential interlock	• SeqILck = 1: All the sequential interlock conditions are operational. • SeqILck = 0: All the sequential interlock conditions are not operational, trips the motor and generates an alarm. Applicable in Auto mode only.

Parameter	Type	Description	Details
ProcILck	BOOL	Process interlock	• ProcILck = 1: All the process interlock conditions are operational. • ProcILck = 0: All the process interlock conditions are not operational, trips the motor and generates an alarm. Applicable in Auto and Manual modes only.
Avail	BOOL	Availability of motor	• Avail = 1: Motor is electrically available. • Avail = 0: Motor is electrically not available. **Stop** command to the motor, OP = 0. The signal comes from the MCC outgoing circuit (electrical protection active, circuit breaker switched off, drawer out, and so on).
OverLoad	BOOL	Overload	• OverLoad = 1: Motor is not in overload condition. • OverLoad = 0: Motor is in overload condition. **Stop** command to the motor, OP = 0. The signal comes from the MCC outgoing circuit (electrical protection active).
Torque	BOOL	Torque	Torque signal.
FbFwd	BOOL	Forward run feedback of motor - electrical	Once the motor is given a **Start** command from output OP1, its electrical run feedback is continuously monitored. This signal comes from the MCC outgoing circuit. If OP1 = 1 and FbFwd = 0, then **Stop** command to the motor, OP1 = 0 is given and an alarm is generated after time delay. If OP1 = 0 and FbFwd = 1, an alarm is generated after time delay.

Parameter	Type	Description	Details
FbRev	BOOL	Reverse run feedback of motor - electrical	Once the motor is given a **Start** command from output OP2, its electrical run feedback is continuously monitored. This signal comes from the MCC outgoing circuit. If OP2 = 1 and FbRev = 0, then **Stop** command to the motor, OP2 = 0 is given and an alarm is generated after time delay. If OP2 = 0 and FbRev = 1, an alarm is generated after time delay.
FbPos1	BOOL	Position feedback 1	Once the actuator is given a **Start** command in forward direction from output OP1, its position feedback is continuously monitored. If Pos1 = 1: **Stop** command to the actuator, OP1 = 0 is given. If OP1 = 1 and FbPos1 = 0 for more than time delay TFbPos, an alarm is generated.
FbPos2	BOOL	Position feedback 2	Once the actuator is given a **Start** command in forward direction from output OP2, its position feedback is continuously monitored. If Pos2 = 1: **Stop** command to the actuator, OP2 = 0 is given. If OP2 = 1 and FbPos2 = 0 for more than time delay TFbPos, an alarm is generated.
ExtSP	REAL	External set point	External set point
PV	REAL	Process value	Current position

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

The AlmInh parameter to inhibit alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the AnlgAct block:

Parameter	Type	Description	Details
AlmInh	BOOL	Alarm inhibition	• AlmInh = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • AlmInh = 0: Alarms are not inhibited.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

To reset, alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the AnlgAct block:

Parameter	Type	Description	Details
AlmRst	BOOL	Reset	• AlmRst = 1: Resets all the alarms generated by the block. • AlmRst = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from GRst pin of GrpCtrl block and resets the trip alarm.

The following table describes the output parameters of the AnlgAct block:

Parameter	Type	Description	Details
Auto	BOOL	Auto mode	Auto = 1: Auto mode is selected. Motor operation is group based on GStart and GStop.
Manual	BOOL	Manual mode	Manual = 1: Manual mode is selected. Motor operation is based on commands (**Start**) and (**Stop**) given by operator from HMI.
Local	BOOL	LocalManual mode	Local = 1: Local mode is selected. Motor operation is based on commands LStart and LStop given from field.

Parameter	Type	Description	Details
Oos	BOOL	Out of Service mode	Oos = 1: Out of Service mode is selected. Motor is not operated and OP = 0.
GActive	BOOL	Group active	The motor is started and is active in the group. Start command pulse (**Start**) is obtained, GActive = 1.
OP1	BOOL	**Output** command to run motor in forward direction	OP1 = 1: Motor is given command to close.
OP2	BOOL	**Output** command to run motor in reverse direction	OP2 = 1: Motor is given command to open.
SP	REAL	Final set point	In Auto mode, SP = ASP. In Manual mode, SP = MSP.
SeqStart	BOOL	**Start** command to next motor in group	Start command (**pulse**) is received and outputs OP1, OP2 and Run = 1, sequential start command (**pulse**) is given to the next motor connected in the group.
SeqStop	BOOL	**Stop** command to next motor in group	Stop or QStop= 1 and OP1, OP2 and Run = 1, sequential stop command is given to the next motor connected in the group.
Dev	REAL	Deviation	Dev = SP - PV.
ILck	BOOL	Interlock status	ILck = 0: All the interlocks are not operational, active interlock is present. ILck = 1: All the interlocks are operational.
Ready	BOOL	Motor ready to start	• Ready = 1: All the conditions to start the motor are operational. • Ready = 0: All the conditions to start the motor are not operational. Motor will not be given **Start** command.

Parameter	Type	Description	Details
Run	BOOL	Motor running	Motor is running. This output is given if the motor is running and run forward feedback is received. Applicable in all the modes of opeartion.
Warn	BOOL	Alert	• Warn = 1: The motor is operational with no active alert present. • Warn = 0: The motor is operational with an active alert present.
Trip	BOOL	Trip	• Trip = 1: The motor is operational with no active trip alarm present. • Trip = 0: The motor has tripped with an active trip alarm present.

The following table describes the input/output parameter of the AnlgAct block:

Parameter	Type	Description
Stx	AnlgAct_St	Status and control of the analog actuator. Refer to the Structure to interface with the HMI *(see page 313)*.

AnlgAct - Representation as SCoD in UAG

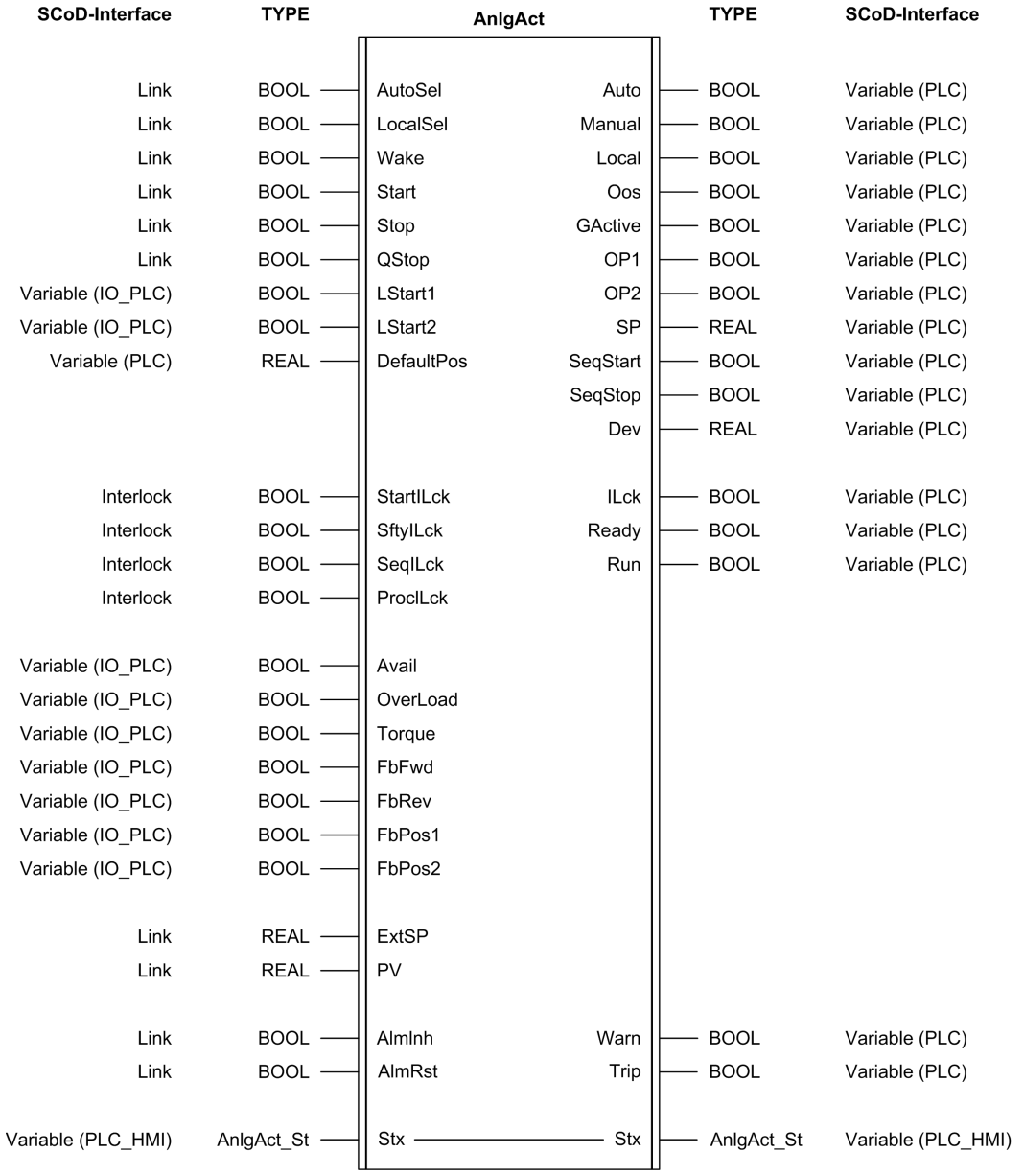
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the Stx variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type HMI is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the `AnlgAct` block.



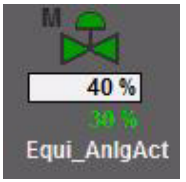
SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Display format (time)	Stx.RunHr	#####	**Configuration**	1	Running hours of motor
Run fb delay (sec)	Stx.TRunFb	0	**Configuration**	2	Sets run feedback delay (s)
Calculation time (sec)	Stx.TC	0	**Configuration**	3	Calculation time
Min pulse length (sec)	Stx.MinPL	0	**Configuration**	4	Sets minimum pulse length
Ramp delay (sec)	Stx.TRamp	0	**Configuration**	5	Ramp time
Constant value	Stx.K	0.0	**Configuration**	6	Constant value
Display format (SP)	Stx.SP	###.##	**Configuration**	7	Set point (RPM)
Inverse direction delay (sec)	Stx.TInv	0	**Configuration**	8	Sets inverse direction start delay (s)
Movement time (sec)	Stx.TMov	0	**Configuration**	9	Movement time
Maximum deviation time (sec)	Stx.TMaxDev	0	**Configuration**	10	Sets maximum deviation time (s)
Measurement unit	Stx.SP	0	**Configuration**	11	Sets measurement unit
Default Position	DefaultPos	0.0	**Configuration1**	1	Default position
Minimum deviation	Stx.MinDev	0.0	**Configuration1**	2	Sets minimum deviation
Divisor constant	Stx.K	0.0	**Configuration1**	3	Constant value
Ramp	Stx.Ramp	0.0	**Configuration1**	7	Ramp value
Maximum deviation	Stx.MaxDev	0.0	**Configuration1**	8	Sets maximum deviation
High limit value	Stx.SPH	100.0	**Limit**	7	Speed set point high limit (RPM)
Low limit value	Stx.SPL	0.0	**Limit**	1	Speed set point low limit (RPM)
Eng. Min	Stx.SP	0	**Range**	1	Sets minimum Eng. value
Eng. Max	Stx.SP	0	**Range**	7	Sets maximum Eng. value
Area Information	AreaInfo	Not available	**Device Data**	1	Area information
Asset ID	AssetID	Not available	**Device Data**	7	Asset ID

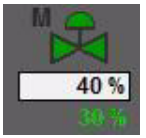
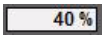
AnlgAct - Representation in the HMI

Analog Actuator (Genie)

The AnlgAct block is represented in the HMI by the following genie.



Genie Elements

Symbol	Meaning
	Opens the AnlgAct popup.
	Displays the set point value of the analog actuator.
	Displays the processed value of the analog actuator.
	The instance name of the AnlgAct block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.
	Genie for AnlgAct as damper.

Animation Description of the Genie

Symbol	Animation	Meaning
	Line color: Steady light gray Fill color: Steady light gray	Analog actuator is in Sleep mode.
	Line color: Steady dark gray Fill color: Steady white	Analog actuator is awake or Local or Manual mode, valve or gate is in closed position.

Symbol	Animation	Meaning
	Line color: Flashing yellow Fill color: Steady white	Analog actuator is awake or Local or Manual mode, valve or gate is in closed position, and an alert alarm is active.
	Line color: Steady yellow Fill color: Steady white	Analog actuator is awake or Local or Manual mode, valve or gate is in closed position, and an alert alarm is acknowledged.
	Line color: Steady dark gray Fill color: Steady green	Analog actuator is awake or Local or Manual mode, and valve or gate is in open position.
	Line color: Flashing yellow Fill color: Steady green	Analog actuator is awake or Local or Manual mode, valve or gate is in opened position, and an alert alarm is active.
	Line color: Steady yellow Fill color: Steady green	Analog actuator is awake or Local or Manual mode, valve or gate is in opened position, and an alert alarm is acknowledged.
	Line color: Steady dark gray Fill color: Flashing red	Analog actuator is tripped, and a trip alarm is active.
	Line color: Steady dark gray Fill color: Steady red	Analog actuator is tripped, and a trip alarm is acknowledged.

Modes

On the top left corner of the motor actuator genie, the mode of operation is indicated.

Mode	Meaning
A	Analog actuator is operating in the Auto mode.
M	Analog actuator is operating in the Manual mode.
L	Analog actuator is operating in the Local mode.
O	Analog actuator is operating in the Out of Service mode.

Symbol	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Alarm Masking

The following table describes the animations for the genie when the alarms are masked:

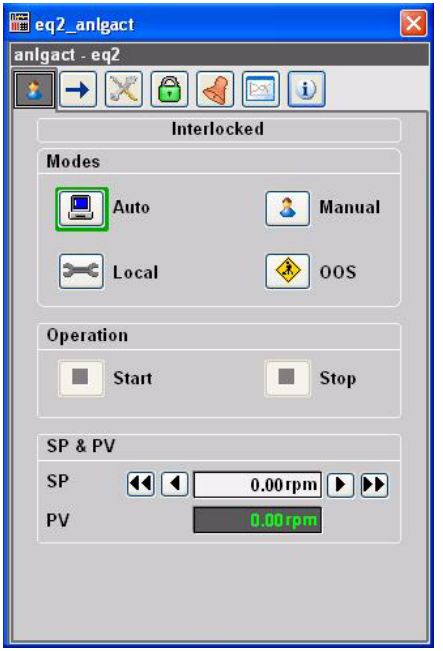
Symbol	Meaning
	The trip or alert alarms in the analog actuator are masked.

Popup

The `AnlgAct` block popup has the following tabs:

- **Operator** tab
- **Extended** tab
- **Configuration** tab
- **Interlock** tab
- **Alarm** tab
- **Trend** tab
- **Information** tab

Operator Tab



This tab has 4 sub-sections:

- **Status**
- **Modes**
- **Operation**
- **SP & PV**

The sub-section **Status** displays the current status and the possible displays are listed as follows:

- **Sleeping**
- **Group Active**
- **Ready**
- **Not Ready**
- **Running**
- **Interlocked**
- **Warning**
- **Trip**

The following table describes the sub-section **Modes**:

Item	Description
Auto	The motor runs in Auto mode controlled by PLC program.
Manual	The motor runs in Manual mode controlled by HMI.
Local	The motor runs in Local mode controlled by local control box of the motor.
OOS	The motor runs in Out of Service mode for maintenance purpose of the object (applicable in all modes).

The following table describes the sub-section **Operation**:

Item	Description
Start	**Start** command is given to actuator to reach position 1.
Stop	**Stop** command is given to actuator to reach position 2.

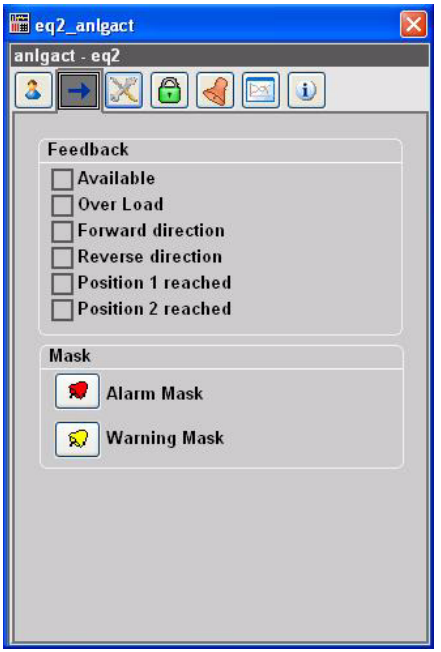
The following table describes the sub-section **SP & PV**:

Item	Description
SP	Displays the set point value (SP) of motor in RPM.
PV	Displays the process value (PV) of motor in RPM.

Symbol	Meaning
	Set point value is ramped down by 1 step.
	Set point value is ramped down to low limit.

Symbol	Meaning
	Set point value is ramped up by 1 step.
	Set point value is ramped up to high limit.

Extended Tab



This tab has 2 sub-sections:

- **Feedback**
- **Mask**

The following table describes the sub-section **Feedback**:

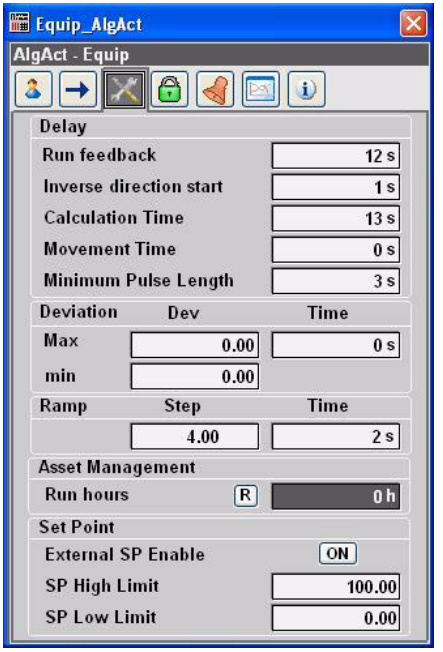
Item	Animation	Description
Available	Steady green	Motor electrically available feedback is obtained.
	Steady gray	Motor electrically not available feedback is obtained.
Over Load	Steady green	Motor overload feedback is obtained.
	Steady gray	Motor overload feedback is not obtained.

Item	Animation	Description
Forward direction	Steady green	Motor running in forward direction, feedback is obtained.
	Steady gray	Motor running in forward direction, feedback is not obtained.
Reverse direction	Steady green	Motor running in reverse direction, feedback is obtained.
	Steady gray	Motor running in reverse direction, feedback is not obtained.
Position 1 reached	Steady green	Position 1 reached, feedback is obtained.
	Steady gray	Position 1 reached, feedback is not obtained.
Position 2 reached	Steady green	Position 2 reached, feedback is obtained.
	Steady gray	Position 2 reached, feedback is not obtained.

The following table describes the sub-section **Mask**:

Item	Description
Alarm Mask	Masks the indicated current trip alarms of this block in the alarm list.
Warning Mask	Masks the current alerts of this block in the alarm list.

Configuration Tab



This tab has 4 sub-sections:

- **Delay**
- **Deviation**
- **Ramp**
- **Asset Management**
- **Set Point**

The following table shows the sub-section **Delay**:

Item	Description
Run feedback	Displays run feedback delay of motor, configurable from HMI. After starting the motor, if run feedback is not received before delay time, an alarm is generated, and motor is tripped.
Inverse direction start	Inverse direction start delay of motor, configurable from HMI. Time taken to switch on motor from forward to reverse direction or vice versa exceeds delay time, an alarm is generated, and motor is tripped.
Calculation Time	Displays the calculation time.

Item	Description
Movement Time	Displays the movement time.
Minimum Pulse Length	Displays the minimum pulse length.

The following table shows the sub-section **Deviation**:

Item	Description
Max Dev	Displays the maximum deviation value, configurable from HMI.
Max Time	Displays the maximum time value, configurable from HMI.
Min Dev	Displays the minimum deviation value, configurable from HMI.

The following table shows the sub-section **Ramp**:

Item	Description
Ramp Step	Displays the ramp step value.
Ramp Time	Displays the ramp time value.

The following table shows the sub-section **Asset Management**:

Item	Description
Run hours	Displays the run hours of motor.
	Resets the run hours value of motor to 0.

The following table shows the sub-section **Set Point**:

Item	Description
External SP Enable	Enables the external set point.
SP High Limit	Displays the set point high limit.
SP Low Limit	Displays the set point low limit.

Interlock Tab

This tab has a sub-section called as **Interlock**, which is described in the following table:

Item	Animation	Description
Starting Interlock	Steady green	**Starting Interlock** is active, and the related device is not blocked.
	Steady gray	**Starting Interlock** is not active, and the related device is blocked.
Safety Interlock	Steady green	**Safety Interlock** is active, and the related device is not blocked.
	Steady gray	**Safety Interlock** is not active, and the related device is blocked.
Sequential Interlock	Steady green	**Sequential Interlock** is active, and the related device is not blocked.
	Steady gray	**Sequential Interlock** is not active, and the related device is blocked.
Process Interlock	Steady green	**Process Interlock** is active, and the related device is not blocked.
	Steady gray	**Process Interlock** is not active, and the related device is blocked.

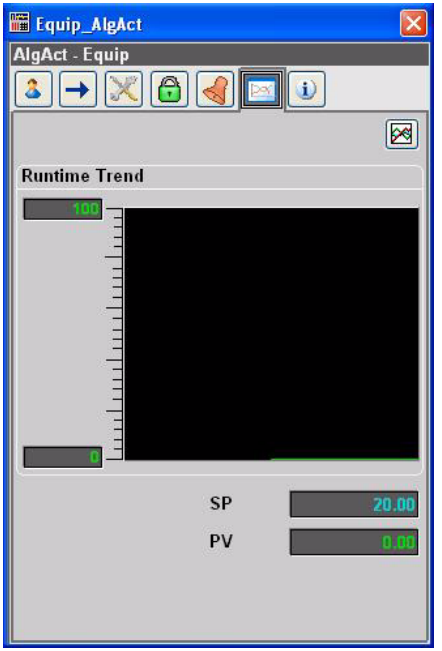
This tab has sub-tabs Interlock, **STI**, **SI**, and **SQI**.

For description about **Interlock** sub-tabs, refer to Commonly Used Tabs *(see page 53)*.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Trend Tab



The sub-section **Runtime Trend** displays the run time trend of the parameters in the control module.

Item	Description
SP	Displays the run time trend of set point.
PV	Displays the run time trend of process variable.

Symbol	Meaning
	Displays the historical trend page.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

AnlgAct_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type AnlgAct_St for the AnlgAct block:

Element	Type	Description
ExtSP	REAL	External set point
ManSP	REAL	Manual set point
PV	REAL	Process value in rpm
SP	REAL	Set point
SPH	REAL	Set point high limit
SPL	REAL	Set point low limit
MaxDev	REAL	Maximum deviation
MinDev	REAL	Minimum deviation
Ramp	REAL	Ramp value
K	REAL	Divisor constant
RunHr	INT	Run hours of motor
TRunFb	INT	Run feedback delay
TInv	INT	Inverse direction start delay
TRamp	INT	Ramp time
TMaxDev	INT	Maximum deviation time
MinPL	INT	Minimum pulse length
TC	INT	Calculation time
Tmov	INT	Movement time
Dummy1	INT	Not used
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Stw2	WORD	Status word 2 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI
ExtSPEn	BOOL	External set point enable

Bit description for the Ctw element:

Bit	Description
0	Auto mode selection from HMI
1	Manual mode selection from HMI

Bit	Description
2	Local mode selection from HMI
3	Out of Service mode selection from HMI
4	Start command from HMI
5	Start2
6	Stop command from HMI
7	Alarm reset from HMI
8	Run hour reset from HMI
9	Not used
10	Rampup/Inching
11	Ramp down
12	Ramp maximum
13	Ramp minimum
14	Alert mask
15	Alarm mask

Bit description for the `Stw1` element:

Bit	Description
0	Auto mode selection
1	Local mode selection
2	Group wake
3	Not used
4	Start interlock
5	Safety interlock
6	Sequential interlock
7	Process interlock
8	Motor electrically available
9	Motor overload
10	Torque
11	Run feedback forward direction
12	Run feedback reverse direction
13	Process feedback/position 1 feedback
14	Alarm inhibition
15	Position 2 feedback

Bit description for the `Stw2` element:

Bit	Description
0	Auto mode active status
1	Manual mode active status
2	Local mode active status
3	Out of Service mode active status
4	Output `OP1` active
5	Output `OP2` active
6	Motor ready status
7	Motor running status
8...11	Not used
12	Interlock active status
13	Group active status
14	Alert active status
15	Trip active status

Bit description for the `Alw1` element:

Bit	Description
0	Motor locally stop
1	Safety interlock not available
2	Sequential interlock not available
3	Process interlock not active
4	Motor electrically not available
5	Motor overload
6	Motor torque not available
7	Run feedback not active when motor is running
8	Run feedback active when motor is not running
9	Reverse feedback not active when motor is running
10	Reverse feedback active when motor is not running
11 ... 12	Not used
13	Maximum deviation value exceeded
14	Run hours maximum value reached
15	Active trip alarm

8.11 MotBasic: Motor Basic

Overview

This section describes the MotBAsic block.

What's in this Section?

This section contains the following topics:

Topic	Page
MotBasic - Description	317
MotBasic - Representation as SCoD in UAG	324
MotBasic - Representation in the HMI	326
MotBasic_St Derived Data Type	332

MotBasic - Description

General

The MotBasic block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB MotBasic
Vijeo Citect	Genie MotBasic
UAG	SCoD MotBasic

Function Description

The MotBasic block is used to control electrical motors with 1 direction of rotation. This block supports only Auto mode of operation.

Auto Mode Operation

A wake-up command (**Wake**) is given by the GrpCtrl block to wake-up the motor from sleep. A group start-up pulse command (**Start**) is given either from the GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group. This sets the output GActive as HIGH if all the interlocks (StartILck, SftyILck, SeqILck, ProcILck) and all the feedbacks (Lstop, Avail, OverLoad, EStop) are HIGH. This sets the output (OP) as HIGH.

Ready output (Ready) is set HIGH when all interlocks and conditions are HIGH.

Interlock output (ILck) is set HIGH when inputs starting interlock (StartILck), safety interlock (SftyILck), sequential interlock (SeqILck) and process interlock (ProcILck) are set HIGH.

After the start command (**Start**) is issued and output (OP) is set HIGH within the specified run feedback delay, the electrical run feedback (FbRun) has to be obtained. This electrical run feedback is continuously monitored by the motor block. If it is not operational, motor is tripped, and an alarm is generated.

After the start command (**Start**) is issued and motor is operational if run feedback (FbRun) is not available for 500 ms, the motor is tripped, and an alarm is generated.

The start command (**Start**) is issued, and output (OP) is set HIGH when the run feedback (FbRun) is obtained. Run output (Run) is set HIGH, and pulse command (**SeqStart**) is given to the next device to start.

Group stop command (**Stop**) or group quick stop command (**QStop**) sets the output command (**OP**) as LOW. Group stop command (**Stop**) is received by the motor from the GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group. Group quick stop command (**QStop**) is received by the motor from the GrpCtrl block.

The stop command (**Stop**) is issued and output (OP) is set LOW when the run feedback (FbRun) is set LOW. Run output (Run) is set LOW and command (**SeqStop**) is given to the next device to stop.

The quick stop command (**QStop**) is issued and output (OP) is set LOW and command (**SeqStop**) is given to the next device to stop.

Interlock Types

There are 4 types of interlocks. The result of individual interlock condition blocks is connected to these interlock pins:

Interlock Name	Parameter	Description
Starting Interlock	StartILck	These interlocks are valid before motor start-up only.
Safety Interlock	SftyILck	All the safety conditions are taken into consideration. This input is active in all the modes.
Sequential Interlock	SeqILck	All the sequential conditions are taken into consideration. This input is active only in Auto mode.
Process Interlock	ProcILck	All the process conditions are taken into consideration. This input is active in Auto and Manual mode.

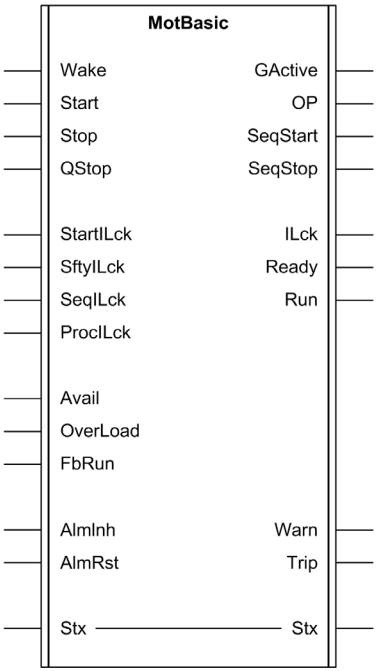
The status of the interlock is displayed in the popup. Detailed description of the interlocks is displayed in the **Interlock** tab of the popup.

The availability (Avail) of the motor is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated. The overload feedback (OverLoad) of the motor is continuously monitored. If motor is started and it is not operational, the motor is tripped, and an alarm is generated.

The alert and trip are indicated in the SCADA. The number of active alarms is shown in the popup of the motor. The details of the active alarms are shown in the **Alarm** tab of the popup. Based on configured conditions, the alarms for this block can be inhibited by connecting the logic to the input pin AlmInh. The input pin AlmRst that is connected from GrpCtrl block or the operator from popup can reset the trip alarms.

Representation in FBD

The following figure shows the FBD representation of the `MotBasic` block.



Parameter Description

The following table describes the input parameters of the `MotBasic` block:

Parameter	Type	Description	Details
`Wake`	`BOOL`	Group wake-up	Group wake-up and request for group availability check: ● `Wake` = 1: Group is active. ● `Wake` = 0: Group is in Sleep mode. This input is connected from output `GWake` of `GrpCtrl`.
`Start`	`BOOL`	Group start	**Start** command to the motor. `Start` = 1: `OP` pin is set to 1. This input is connected from the `GStart` pin of `GrpCtrl` or `GrpStrtMgt` or `RouteMgt` or previous device in the group.

Parameter	Type	Description	Details
Stop	BOOL	Group stop	**Stop** command to the motor. Stop = 1: OP pin is set to 0. This input is connected from the GStop pin of GrpCtrl or GrpStpMgt or RouteMgt or previous device in the group.
QStop	BOOL	Group quick stop	**Stop** command to the motor. QStop = 1: OP pin is set to 0. This input is connected from the GQStop pin of GrpCtrl.
StartILck	BOOL	Starting interlock	• StartILck = 1: All the starting interlock conditions are operational. • StartILck = 0: All the starting interlock conditions are not operational. Applicable only before starting of the motor. After motor is started, StartILck is not considered.
SftyILck	BOOL	Safety interlock	• SftyILck = 1: All the safety interlock conditions are operational. • SftyILck = 0: All the safety interlock conditions are not operational, trips the motor and generates an alarm.
SeqILck	BOOL	Sequential interlock	• SeqILck = 1: All the sequential interlock conditions are operational. • SeqILck = 0: All the sequential interlock conditions are not operational, trips the motor and generates an alarm.
ProcILck	BOOL	Process interlock	• ProcILck = 1: All the process interlock conditions are operational. • ProcILck = 0: All the process interlock conditions are not operational, trips the motor and generates an alarm.

Parameter	Type	Description	Details
Avail	BOOL	Availability of motor	• Avail = 1: Motor is electrically available. • Avail = 0: Motor is electrically not available. **Stop** command to the motor, OP = 0. The signal comes from the MCC outgoing circuit (electrical protection active, circuit breaker switched off, drawer out, and so on).
OverLoad	BOOL	Overload	• OverLoad = 1: Motor is not in overload condition. • OverLoad = 0: Motor is in overload condition. **Stop** command to the motor, OP = 0. The signal comes from the MCC outgoing circuit (electrical protection active).
FbRun	BOOL	Run feedback of motor - electrical	Once the motor is given a **Start** command from output OP, its electrical run feedback is continuously monitored. This signal comes from the MCC outgoing circuit. If OP = 1 and FbRun = 0, then **Stop** command to the motor, OP = 0 is given and an alarm is generated after time delay. If OP = 0 and FbRun = 1, an alarm is generated after time delay.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

The AlmInh parameter to inhibit alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the MotBasic block:

Parameter	Type	Description	Details
AlmInh	BOOL	Alarm inhibition	• AlmInh = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • AlmInh = 0: Alarms are not inhibited.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

To reset, alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the MotBasic block:

Parameter	Type	Description	Details
AlmRst	BOOL	Reset	• AlmRst = 1: Resets all the alarms generated by the block. • AlmRst = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from GRst pin of GrpCtrl block and resets the trip alarm.

The following table describes the output parameters of the MotBasic block:

Parameter	Type	Description	Details
GActive	BOOL	Group active	The motor is started in the group and is active in the group. Start command pulse **Start** is obtained, GActive = 1.
OP	BOOL	**Output** command to run motor in forward direction	• OP = 1: Motor is given command to start. • OP = 0: Motor is given command to stop. The output switches on the contactor or circuit breaker controlling the motor in forward direction.

Parameter	Type	Description	Details
SeqStart	BOOL	**Start** command to next motor in group	Start command (pulse) is received and outputs OP and Run = 1, sequential start command pulse is given to the next motor connected in the group.
SeqStop	BOOL	**Stop** command to next motor in group	Stop or QStop = 1 and OP and Run = 0, sequential stop command is given to the next motor connected in the group.
ILck	BOOL	Interlock status	• ILck = 1: All the interlocks are operational. • ILck = 0: All the interlocks are not operational, active interlock is present.
Ready	BOOL	Motor ready to start	• Ready = 1: All the conditions to start the motor are operational. • Ready = 0: All the conditions to start the motor are not operational. Motor will not be given **Start** command.
Run	BOOL	Motor running	Motor is running. This output is given if the motor is running and run feedback (FbRun) is received. Applicable in all the modes of opeartion.
Warn	BOOL	Alert	• Warn = 1: The motor is operational with no active alert present. • Warn = 0: The motor is operational with an active alert present.
Trip	BOOL	Trip	• Trip = 1: The motor is operational with no active trip alarm present. • Trip = 0: The motor has tripped with an active trip alarm present.

The following table describes the input/output parameter of the MotBasic block:

Parameter	Type	Description
Stx	MotBasic_St	Status and control of the motor. Refer to the Structure to interface with the HMI *(see page 332)*.

MotBasic - Representation as SCoD in UAG

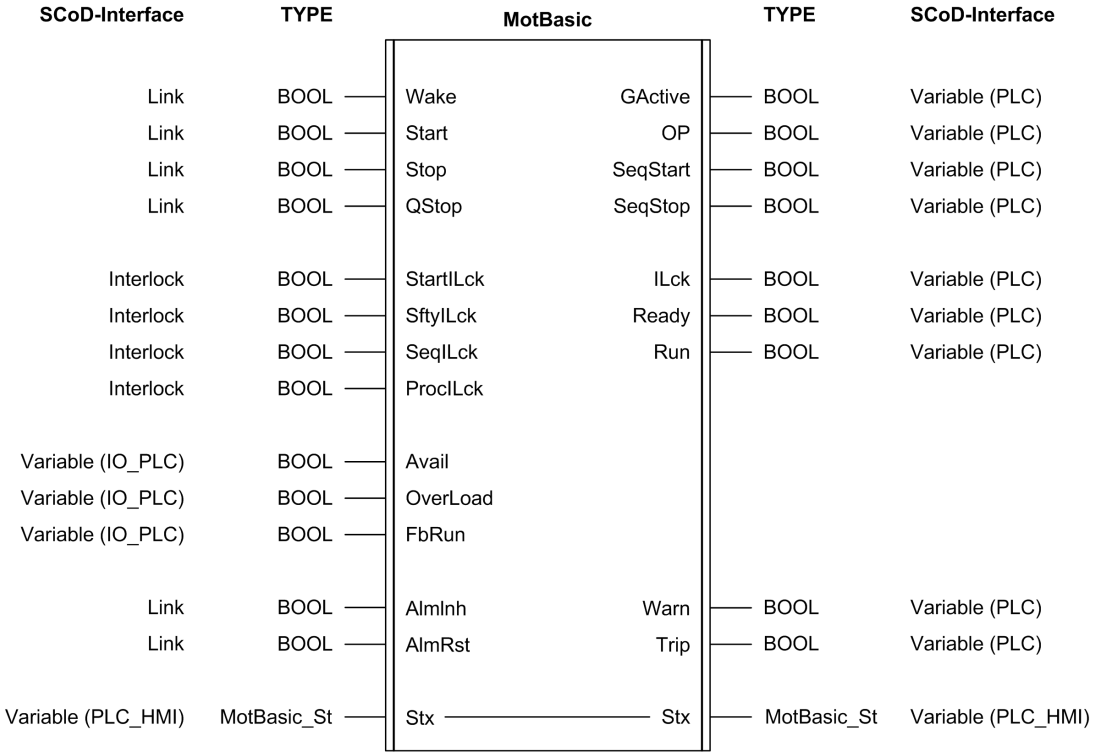
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the Stx variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type HMI is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the MotBasic block.



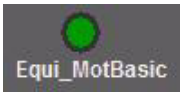
SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Display format (time)	`Stx.TRunFB`	#####	**Configuration**	1	Run feedback delay
Run fb delay (sec)	`Stx.TRunFB`	0	**Configuration**	7	Run feedback delay
Area Information	`AreaInfo`	Not available	**Device Data**	1	Area information
Asset ID	`AssetID`	Not available	**Device Data**	7	Asset ID

`MotBasic` - Representation in the HMI

Motor Basic (Genie)

The `MotBasic` block is represented in the HMI by the following genie.



Genie Elements

Symbol	Meaning
	Opens the `MotBasic` popup.
	The instance name of the `MotBasic` block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

Animation Description of the Genie

Symbol	Animation	Meaning
	Line color: Steady light gray Fill color: Steady light gray	Motor is in Sleep mode.
	Line color: Steady dark gray Fill color: Steady white	Motor is awake or in Local mode or Manual mode and ready. Motor is awake or in Local mode or Manual mode and ready after trip alarms are reset.
	Line color: Dark gray Fill color: Light gray	Motor is awake or in Local mode or Manual mode and not ready.
	Line color: Steady dark gray Fill color: Flashing green	Motor is started, and feedback is yet to be obtained or motor is stopping.
	Line color: Steady dark gray Fill color: Steady green	Motor is running.
	Line color: Flashing yellow Fill color: Steady green	Motor is running, and an alert alarm is active.
	Line color: Steady yellow Fill color: Steady green	Motor is running, and an alert alarm is acknowledged.

Symbol	Animation	Meaning
	Line color: Steady dark gray Fill color: Flashing red	Motor is tripped, and a trip alarm is active.
	Line color: Steady dark gray Fill color: Steady red	Motor is tripped, and a trip alarm is acknowledged.

Symbol	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Alarm Masking

The following table describes the animations for the genie when the alarms are masked:

Symbol	Meaning
	The trip or alert alarms in the motor basic are masked.

Popup

The `MotBasic` block popup has the following tabs:

- **Operator** tab
- **Configuration** tab
- **Interlock** tab
- **Alarm** tab
- **Information** tab

Operator Tab

This tab has 3 sub-sections:

- **Status**
- **Feedback**
- **Mask**

The sub-section **Status** displays the current status and the possible displays are listed as follows:

- **Sleeping**
- **Group Active**
- **Ready**
- **Not Ready**
- **Starting**
- **Stopping**
- **Running**
- **Interlocked**
- **Warning**
- **Trip**

The following table describes the sub-section **Feedback**:

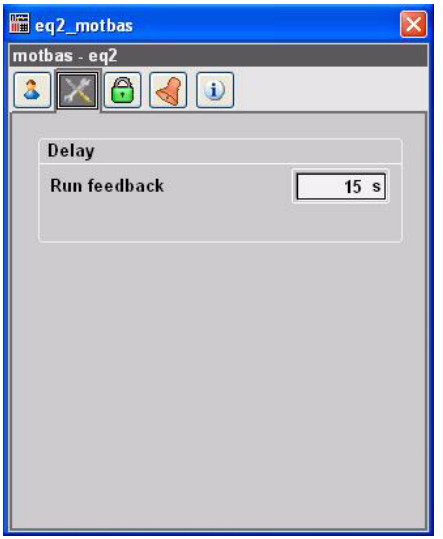
Item	Animation	Description
Available	Steady green	Motor electrically available feedback is obtained.
	Steady gray	Motor electrically not available feedback is obtained.
Over Load	Steady green	Motor overload feedback is obtained.
	Steady gray	Motor overload feedback is not obtained.

Item	Animation	Description
Run	Steady green	Motor running feedback is obtained.
	Steady gray	Motor not running feedback is obtained.

The following table describes the sub-section **Mask**:

Item	Description
Alarm Mask	Masks the indicated current trip alarms of this block in the alarm list.
Warning Mask	Masks the current alerts of this block in the alarm list.

Configuration Tab



This tab has a sub-section called as **Delay**, which is described in the following table:

Item	Description
Run feedback	Displays the run feedback delay of motor, configurable from HMI. After starting the motor, if run feedback is not received before delay time, an alarm is generated, and motor is tripped.

Interlock Tab

This tab has a sub-section called as **Interlock**, which is described in the following table:

Item	Animation	Description
Starting Interlock	Steady green	**Starting Interlock** is active, and the related device is not blocked.
	Steady gray	**Starting Interlock** is not active, and the related device is blocked.
Safety Interlock	Steady green	**Safety Interlock** is active, and the related device is not blocked.
	Steady gray	**Safety Interlock** is not active, and the related device is blocked.
Sequential Interlock	Steady green	**Sequential Interlock** is active, and the related device is not blocked.
	Steady gray	**Sequential Interlock** is not active, and the related device is blocked.
Process Interlock	Steady green	**Process Interlock** is active, and the related device is not blocked.
	Steady gray	**Process Interlock** is not active, and the related device is blocked.

This tab has sub-tabs Interlock, **STI**, **SI**, and **SQI**.

For description about **Interlock** sub-tabs, refer to Commonly Used Tabs *(see page 53)*.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

MotBasic_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type MotBasic_St for the MotBasic block:

Element	Type	Description
TRunFb	INT	Run feedback delay
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Stw2	WORD	Status word 2 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI
Dummy	WORD	Not used

Bit description for the Ctw element:

Bit	Description
0...6	Not used
7	Alarm reset
8...15	Not used

Bit description for the Stw1 element:

Bit	Description
0...1	Not used
2	Group wake
3	Not used
4	Starting interlock
5	Safety interlock
6	Sequential interlock
7	Process interlock
8	Motor electrically available
9	Motor overload
10	Not used
11	Run feedback
12...13	Not used
14	Alarm inhibition
15	Not used

Bit description for the `Stw2` element:

Bit	Description
0...3	Not used
4	Output `OP` active
5	Not used
6	Motor ready status
7	Motor running status
8...11	Not used
12	Interlock active status
13	Group active status
14	Alert active status
15	Trip active status

Bit description for the `Alw1` element:

Bit	Description
0	Not used
1	Safety interlock not available
2	Sequential interlock not available
3	Process interlock not available
4	Motor electrically not available
5	Motor overload
6	Not used
7	Run feedback not active when motor is running
8	Run feedback active when motor is not running
9...14	Not used
15	Active trip alarm

MMM Process

9

Overview

The following chapter describes the MMM process objects.

⚠ WARNING

MISAPPLICATION OF MODULES

The modules in this section do not reflect any specific installation. Before adopting these modules for use in a specific application, the engineer must:

- conduct a safety analysis for the application and equipment installed
- verify that all modules are appropriate for the equipment or function in the installation
- supply appropriate parameters particularly for limits
- check that all sensors and actuators are compatible with the modules selected
- thoroughly test all functions during verification and commissioning
- provide independent paths for critical control functions (emergency stop, over-limit conditions and so on.), according to the safety analysis and applicable codes and regulations.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
9.1	`GrpCtrl`: Group Control	336
9.2	`GrpStrtMgt`: Group Start Management	357
9.3	`GrpStpMgt`: Group Stop Management	371
9.4	`BCnvyr1d`: Belt Conveyor 1 Direction	385
9.5	`RouteMgt`: Route Management	418
9.6	`BagFltr`: Bag Filter Sequence Block	439

9.1 GrpCtrl: Group Control

Overview

This section describes the GrpCtrl block.

What's in this Section?

This section contains the following topics:

Topic	Page
GrpCtrl - Description	337
GrpCtrl - Representation as SCoD in UAG	344
GrpCtrl - Representation in the HMI	347
GrpCtrl_St Derived Data Type	355

GrpCtrl - Description

General

The GrpCtrl block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB GrpCtrl
Vijeo Citect	Genie GrpCtrl
UAG	SCoD GrpCtrl

Function Description

The GrpCtrl block is used to start/stop the devices in the group.

The following table shows the 2 methods to manage the group:

Method	Description
Daisy chain	The GrpCtrl block is directly linked to the device 1 in the group, and the device 1 gives commands to the device 2 after it starts and so on. **NOTE:** This process continues for all devices in the group.
Using management blocks	The GrpCtrl block is connected to the GrpStrtMgt and GrpStpMgt block, and these blocks are in turn connected to the devices (for example, DigIn, Mot1d, DigAct, and so on) in the group.

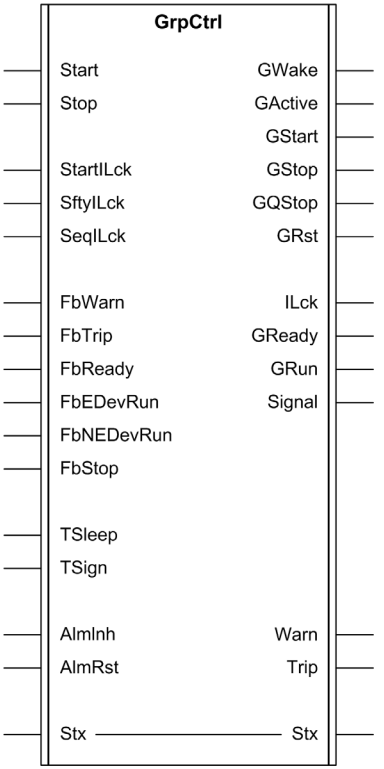
The following table shows the 5 main functions of the `GrpCtrl` block:

Function	Description	Result
Group wake/sleep	Group availability check command given by the operator from HMI wakes up the group by giving wake-up command to all the devices in the group	Group goes to Wake mode and monitors the warning, trip, essential device run, non essential device run, and group stop feedbacks. If there are no interlocks (starting or safety or sequential) and trip alarms, then the group is available or ready to start. After the stop feedback is received, the group goes to sleep mode after a predefined time delay (`TSleep`) until the next **Start** command is issued on the condition that there is no active trip alarm.
Group start and restart	When the group is available, a **Start** or **Restart** command is given by operator from HMI popup or by PLC program through the input pin `Start`.	If the block is ready, **Start** command is given to the devices either by daisy chain (connection to first motor in the sequence) or by using management blocks (connected to `GrpStrtMgt` block). Group blocks generates a signal for the hooter for a predefined time (`Tsign`) before giving the **Start** command to the group depending on the essential devices running feedback. If the group does not receive run feedbacks for more than predefined time(`Stx.TStart`), then an alert is generated.
Group stop	• When the group has to be stopped, a **Stop** command is given by operator from HMI popup or by PLC program through the input pin `Stop`. • When the sequential interlock is not operational, the group stop command is issued.	**Stop** command is given to the devices either by daisy chain (connection to last motor in the sequence) or by using management blocks (connected to `GrpStpMgt` block). If the group does not receive stop feedback for more than predefined time (`Stx.TStop`), then an alert is generated.

Function	Description	Result
Group quick stop	When the group has to be stopped quickly, a quick stop command is given by operator from HMI popup. When the emergency or safety interlock is not operational, a group quick stop command is issued.	Quick stop command is given to all the devices in the group. If the group does not receive stop feedback for more than predefined time (`Stx.TStop`), then an alert is generated.
Reset trip alarms of blocks in the group	When a **Reset** command is issued, the trip alarms of all the devices in the group are reset. **NOTE:** The output pin reset has to be connected to the `AlmRst` input of all the devices, motors, and actuators in the group.	The warning and trip feedback in the block monitors the availability of an active alert or alarm in the group.

Representation in FBD

The following figure shows the FBD representation of the `GrpCtrl` block.



Parameter Description

The following table describes the input parameters of the GrpCtrl block:

Parameter	Type	Description	Details
Start	BOOL	Group start command	Start = 0...1: **Start** command (GStart = 0...1) is given to the group. This is a pulse input.
Stop	BOOL	Group stop command	Stop = 1: **Stop** command (GStop = 1) is given to the group.
StartILck	BOOL	Starting interlock	• StartILck = 1: All the starting interlock conditions are operational. • StartILck = 0: Not all the starting interlock conditions are operational. **NOTE:** Applicable only before starting of the group. After the motor is started, StartILck is not considered. This is used to interlock the group start-up with any other group or with any event.
SftyILck	BOOL	Safety interlock	• SftyILck = 1: All the safety interlock conditions are operational. • SftyILck = 0: Not all the safety interlock conditions are operational. Generates the **Stop** command GQStop = 1.
SeqILck	BOOL	Sequential interlock	• SeqILck = 1: All the sequential interlock conditions are operational. • SeqILck = 0: Not all the sequential interlock conditions are operational. Generates the **Stop** command GQStop = 1.
FbWarn	BOOL	Feedback - alert	FbWarn = 1: Presence of an active alert in the group.
FbTrip	BOOL	Feedback - trip alarm	FbTrip = 1: Presence of an active trip alarm in the group.
FbReady	BOOL	Feedback ready	FbReady = 1: The devices in the group are ready to start.

Parameter	Type	Description	Details
FbEDevRun	BOOL	Run feedback - essential devices	• FbEDevRun = 1: All essential devices in the group are running. • FbEDevRun = 0: Not all essential devices in the group are running.
FbNEDevRun	BOOL	Run feedback - non essential devices	• FbNEDevRun = 1: All non essential devices in the group are running. • FbNEDevRun = 0: Not all non essential devices in the group are running.
FbStop	BOOL	Feedback - stop	FbStop = 1: All devices in the group are stopped.
TSleep	INT	Sleep time	Time to return group to sleep.
TSign	INT	Sign time	Defines the operation time delay of visible and audible alert signals on group start-up.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

The AlmInh parameter to inhibit alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the GrpCtrl block:

Parameter	Type	Description	Details
AlmInh	BOOL	Alarm inhibition	• AlmInh = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • AlmInh = 0: Alarms are not inhibited.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

To reset, all generated alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the GrpCtrl block:

Parameter	Type	Description	Details
AlmRst	BOOL	Reset	• AlmRst = 1: Resets all the alarms generated by the block. • AlmRst = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from GRst pin of GrpCtrl block and resets the trip alarm.

The following table describes the output parameters of the GrpCtrl block:

Parameter	Type	Description	Details
GWake	BOOL	Group wake-up	Group wake-up and request for group availability check: • GWake = 1: Group is in Wake mode. • GWake = 0: Group is in Sleep mode. **NOTE:** This pin is connected to each block in the group.
GActive	BOOL	Group active	Block active in the group.
GStart	BOOL	**Start** command to the group	GStart = 1: **Start** command (pulse output GStart = 0...1) is given to the first motor in the sequence or GrpStrtMgt block.
GStop	BOOL	**Stop** command to the group	GStop = 1: **Stop** command is given to the last motor in the sequence or GrpStpMgt block.
GQStop	BOOL	**Quick stop** command to the group	GQStop = 1: **Stop** command is given to all the devices in the group.
GRst	BOOL	Group reset	Reset of trip alarms. **NOTE:** This output has to be connected to input AlmRst of each block in the group.
ILck	BOOL	Interlock	• ILck = 1: All the interlocks are operational. • ILck = 0: Not all the interlocks are operational. There is an active interlock present.

Parameter	Type	Description	Details
GReady	BOOL	Ready	• GReady = 1: All the conditions to start the group are operational. • GReady = 0: Not all the conditions to start the group are operational. Group will not be given the **Start** command.
GRun	BOOL	Run	• GRun = 1: All the devices in the group are running. • GRun = 0: Not all the devices in the group are running.
Signal	BOOL	Signal	Connection of visible and audible alert signals on group start-up or restart during Tsign time.
Warn	BOOL	Alert	• Warn = 1: The group is operational with no active alert present. • Warn = 0: The group is operational with an active alert present.
Trip	BOOL	Trip	• Trip = 1: The group is operational with no active trip alarm is present. • Trip = 0: The group has tripped with an active trip alarm is present.

The following table describes the input/output parameter of the GrpCtrl block:

Parameter	Type	Description
Stx	GrpCtrl_St	Status and control of the GrpCtrl block. Refer to the Structure to interface with the HMI *(see page 355)*.

GrpCtrl - Representation as SCoD in UAG

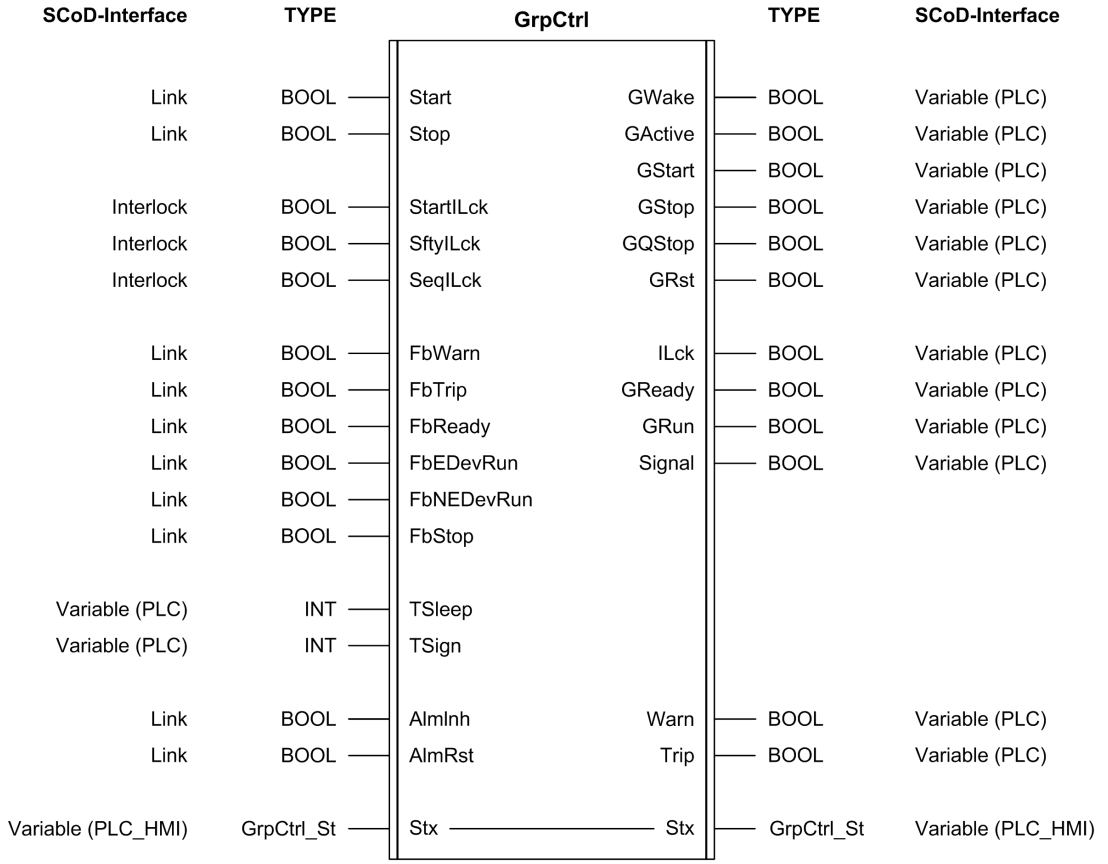
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the Stx variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type HMI is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the `GrpCtrl` block.



SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Display format (time)	`Stx.Tstart`	#####	**Configuration**	1	Sets start-up delay (s).
Group startup delay (sec)	`Stx.TStart`	0	**Configuration**	2	Sets start-up delay (s).
Group stop delay (sec)	`Stx.TStop`	0	**Configuration**	3	Sets shutdown delay (s).

Name	Variable	Initial Value	Property Tab	Position	Description
Time duration for signal (sec)	`TSleep`	0	**Configuration**	8	Sets sleep time (s).
Time to return to sleep (sec)	`TSign`	0	**Configuration**	9	Sets sign time (s).

GrpCtrl - Representation in the HMI

Group Control (Genie)

The GrpCtrl block is represented in the HMI by the following genie.

Genie Elements

Symbol	Meaning
	Opens the GrpCtrl popup.
	The instance name of the GrpCtrl block is visible by default. You can hide/unhide it by clicking the **Hide/Unhide** button in the template.

Animation Description of the Genie

The following table describes the different animations for the **AVAIL** genie element:

Symbol	Animation	Meaning
	Steady gray	Group is in Sleep mode.
	Flashing yellow	Presence of an alert alarm in the group. **NOTE:** Absence of trip alarms or interlocks.
	Steady yellow	Alert alarm in the group is acknowledged.
	Flashing red	Presence of trip alarm or interlock.
	Steady red	Active trip alarm is acknowledged.
	Steady white	• Group is available for start-up or, • Group is available for restart after resetting the trip alarm or interlock.

The following table describes the different animations for the **RUN** genie element:

Symbol	Animation	Meaning
	Steady gray	Group is in Sleep mode.
	Steady white	Group is in Awake mode.
	Upper half flashing green	Group is starting without essential device feedback.
	Lower half flashing green	Group is starting without non essential device feedback.
	Flashing green	Group is starting without essential and non-essential device feedback.
	Steady green	Group is running.

The following table describes the different animations for the **STOP** genie element:

Symbol	Animation	Meaning
	Steady gray	Group is in Sleep mode.
	Flashing white	Group is in Stop mode.
	Steady white	Group stop is complete or group is wake and active.
	Flashing red	Group is stopping in Quick Stop mode.

The following table describes the animations for the genie when the alarms are masked:

Symbol	Animation	Meaning
	Steady gray	The trip alarms or alerts in the `GrpCtrl` block are masked (`AlmInh` = 1).

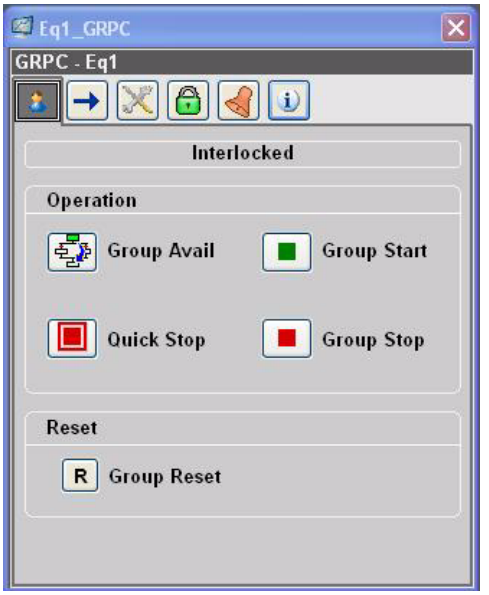
Symbol	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Popup

The `GrpCtrl` block popup has the following tabs:

- **Operator** tab
- **Extended** tab
- **Configuration** tab
- **Interlock** tab
- **Alarm** tab
- **Information** tab

Operator Tab



This tab has 3 sub-sections:

- **Status**
- **Operation**
- **Reset**

The sub-section **Status** displays the current status.

The possible displays are listed as follows:

- **Sleeping**
- **Group Active**
- **Ready**
- **Not Ready**
- **Signal**
- **Starting**
- **Stopping**

- **Running**
- **Interlocked**
- **Warning**
- **Alarm**

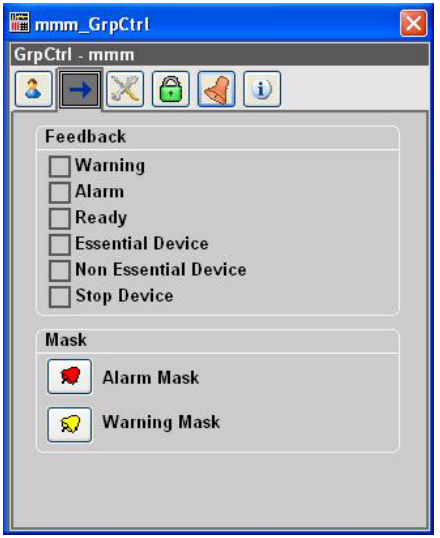
The following table describes the sub-section **Operation**:

Item	Description
Group Avail	Wake-up all devices in the group.
Group Start	Starts all devices in the group.
Quick Stop	Stops all devices in the group simultaneously.
Group Stop	Stops all devices in the group sequentially.

The following table describes the sub-section **Reset**:

Item	Description
Group Reset	Resets the alarms of all devices in the group.

Extended Tab



This tab has 2 sub-sections:

- **Feedback**
- **Mask**

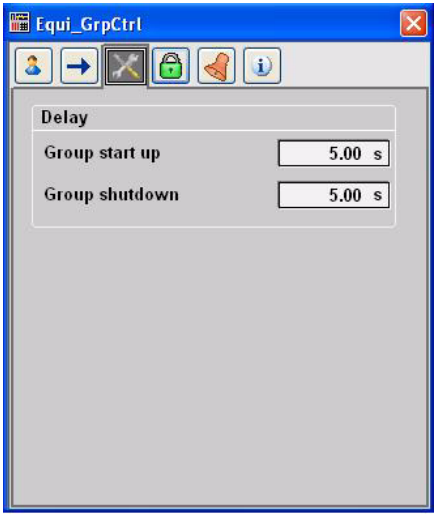
The following table describes the sub-section **Feedback**:

Item	Animation	Description
Warning	Steady green	Alert feedback of group devices is obtained.
	Steady gray	Alert feedback of group devices is not obtained.
Alarm	Steady green	Alarm feedback of group devices is obtained.
	Steady gray	Alarm feedback of group devices is not obtained.
Ready	Steady green	Ready feedback of group devices is obtained.
	Steady gray	Ready feedback of group devices is not obtained.
Essential Device	Steady green	Run feedback of essential devices is obtained.
	Steady gray	Run feedback of essential devices is not obtained.
Non Essential Device	Steady green	Run feedback of non essential devices is obtained.
	Steady gray	Run feedback of non essential devices is not obtained.
Stop Device	Steady green	Stop feedback of group devices is obtained.
	Steady gray	Stop feedback of group devices is not obtained.

The following table describes the sub-section **Mask**:

Item	Description
Alarm Mask	Masks the indicated current trip alarms of this block in the alarm list.
Warning Mask	Masks the current alerts of this block in the alarm list.

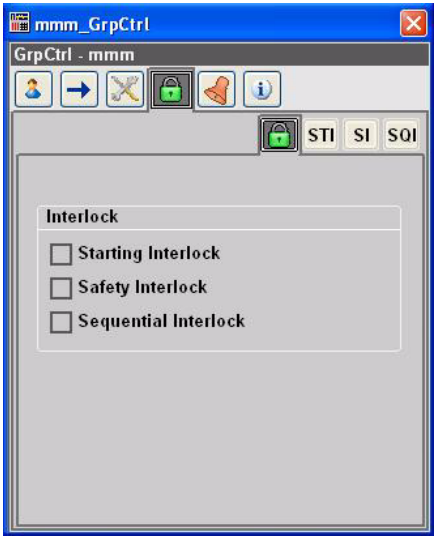
Configuration Tab



This tab has a sub-section called as **Delay**, which is described in the following table:

Item	Description
Group start up	Displays the group start-up delay configurable from HMI. Devices start in the group within the defined delay time.
Group shutdown	Displays the group stop delay configurable from HMI. Devices stop in the group within the defined delay time.

Interlock Tab



This tab has a sub-section called as **Interlock**, which is described in the following table:

Item	Animation	Description
Starting Interlock	Steady green	**Starting Interlock** is active, and the related device is not blocked.
	Steady gray	**Starting Interlock** is not active, and the related device is blocked.
Safety Interlock	Steady green	**Safety Interlock** is active, and the related device is not blocked.
	Steady gray	**Safety Interlock** is not active, and the related device is blocked.
Sequential Interlock	Steady green	**Sequential Interlock** is active, and the related device is not blocked.
	Steady gray	**Sequential Interlock** is not active, and the related device is blocked.

This tab has sub-tabs Interlock, **STI**, **SI**, and **SQI**.

For description about **Interlock** sub-tabs, refer to Commonly Used Tabs *(see page 53)*.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

GrpCtrl_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type GrpCtrl_St for the GrpCtrl block:

Element	Type	Description
TStart	INT	Group start-up monitoring time
TStop	INT	Group stop monitoring time
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Stw2	WORD	Status word 2 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI

Bit description for the Ctw element:

Bit	Description
0	Group availability check
1	Group start-up command
2	Group stop command
3	Group quick stop command
4	Group reset command
5...6	Not used
7	Alarm reset
8...13	Not used
14	Alarm mask
15	Alert mask

Bit description for the Stw1 element:

Bit	Description
0	Start
1	Stop
2	Starting interlock
3	Safety interlock
4	Sequential interlock
5	Feedback - alert
6	Feedback - trip alarm

Bit	Description
7	Feedback - ready
8	Run feedback - essential devices
9	Run feedback - non essential devices
10	Feedback - stop
11...13	Not used
14	Alarm inhibition
15	Not used

Bit description for the `Stw2` element:

Bit	Description
0	Group wake-up
1	Group active
2	Group start
3	Group stop
4	Group quick stop
5	Group ready
6	Group run
7	Signal
8	Group starting
9	Group stopping
10...11	Not used
12	Interlock
13	Not used
14	Alert
15	Trip

Bit description for the `Alw` element:

Bit	Description
0	Safety interlock not active
1	Sequential interlock not active
2	Start-up time exceeded
3	Stop time exceeded
4...14	Not used
15	Group trip

9.2 GrpStrtMgt: Group Start Management

Overview

This section describes the GrpStrtMgt block.

What's in this Section?

This section contains the following topics:

Topic	Page
GrpStrtMgt - Description	358
GrpStrtMgt - Representation as SCoD in UAG	362
GrpStrtMgt - Representation in the HMI	366
GrpStrtMgt_St Derived Data Type	370

GrpStrtMgt - Description

General

The GrpStrtMgt block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB GrpStrtMgt
Vijeo Citect	Genie GrpStrtMgt
UAG	SCoD GrpStrtMgt

Function Description

The GrpStrtMgt block is used to manage the sequential start of the devices in the group.

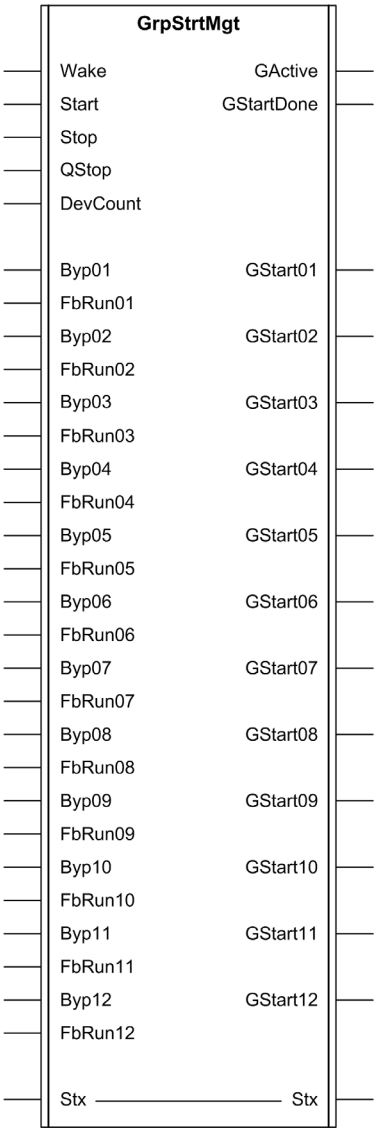
This block is used as an intermediate block between the GrpCtrl and devices (motors/actuators), and can handle a sequential start of up to 12 devices. If there are more than 12 devices, the GrpStrtMgt can be extended by connecting the GStartDone output of this block to the Start pin of the next GrpStrtMgt block.

The status of the devices are available through the feedback device input FbRunx (x: 1...12). Time delay to start the next device is configured from the popup through TStartx (x: 1...12). The popup also shows the remaining time to start the next device TRStartx (x: 1...12). This configured time can be bypassed, and an immediate **Start** command can be provided by assigning specific conditions to the input Bypx (x: 1...12).

After the elapsed time equals with the configured time delay TStartx, the output GStartx (x: 1...12) will give an output. The GStartx output is connected to the Start pin of the device/block. Hence the device/block gets the **Start** command from the GrpStrtMgt block.

Representation in FBD

The following figure shows the FBD representation of the `GrpStrtMgt` block.



Parameter Description

The following table describes the input parameters of the GrpStrtMgt block:

Parameter	Type	Description	Details
Wake	BOOL	Group wake-up	Group wake-up and request for group availability check. • Wake = 1: Group is active. • Wake = 0: Group is in Sleep mode. This input is connected from output GWake of GrpCtrl.
Start	BOOL	**Start** command	**Start** command starts all the devices in the group sequentially.
Stop	BOOL	**Stop** command	Stop = 0 to 1: The block stops sending **Start** command and GActive = 0.
QStop	BOOL	Quick stop command (**QStop**)	QStop = 0 to 1: The block stops sending **Start** command and GActive = 0.
DevCount	INT	Device count	The number of devices are present in the group.
Bypx (x: 01...12).	BOOL	Bypass of timer Tx (x: 01...12)	Bypx (x: 01...12) = 1: Bypass the timer and starts the device x (x: 01...12) immediately.
FbRunx (x: 01...12)	BOOL	Start feedback device x (x: 01...12)	Device x (x: 01...12) is active or started.

The following table describes the output parameters of the GrpStrtMgt block:

Parameter	Type	Description	Details
GActive	BOOL	Group active	The GrpStrtMgt is started in the group and is active in the group.
GStartDone	BOOL	Group start activity is done	This output is set when all the devices in the group are started. It is connected to the Start pin of next GrpStrtMgt block.
GStartx (x: 01...12)	BOOL	**Group Start** command x (x: 01...12)	GStartx (x: 01...12) = 1: Starts device x (x: 01...12). **NOTE:** To be connected to input Start pin of the device x (x: 01...12).

The following table describes the input/output parameter of the GrpStrtMgt block:

Parameter	Type	Description
Stx	GrpStrtMgt_St	Status and control of the GrpStrtMgt block. Refer to the Structure to interface with the HMi *(see page 370)*.

`GrpStrtMgt` - Representation as SCoD in UAG

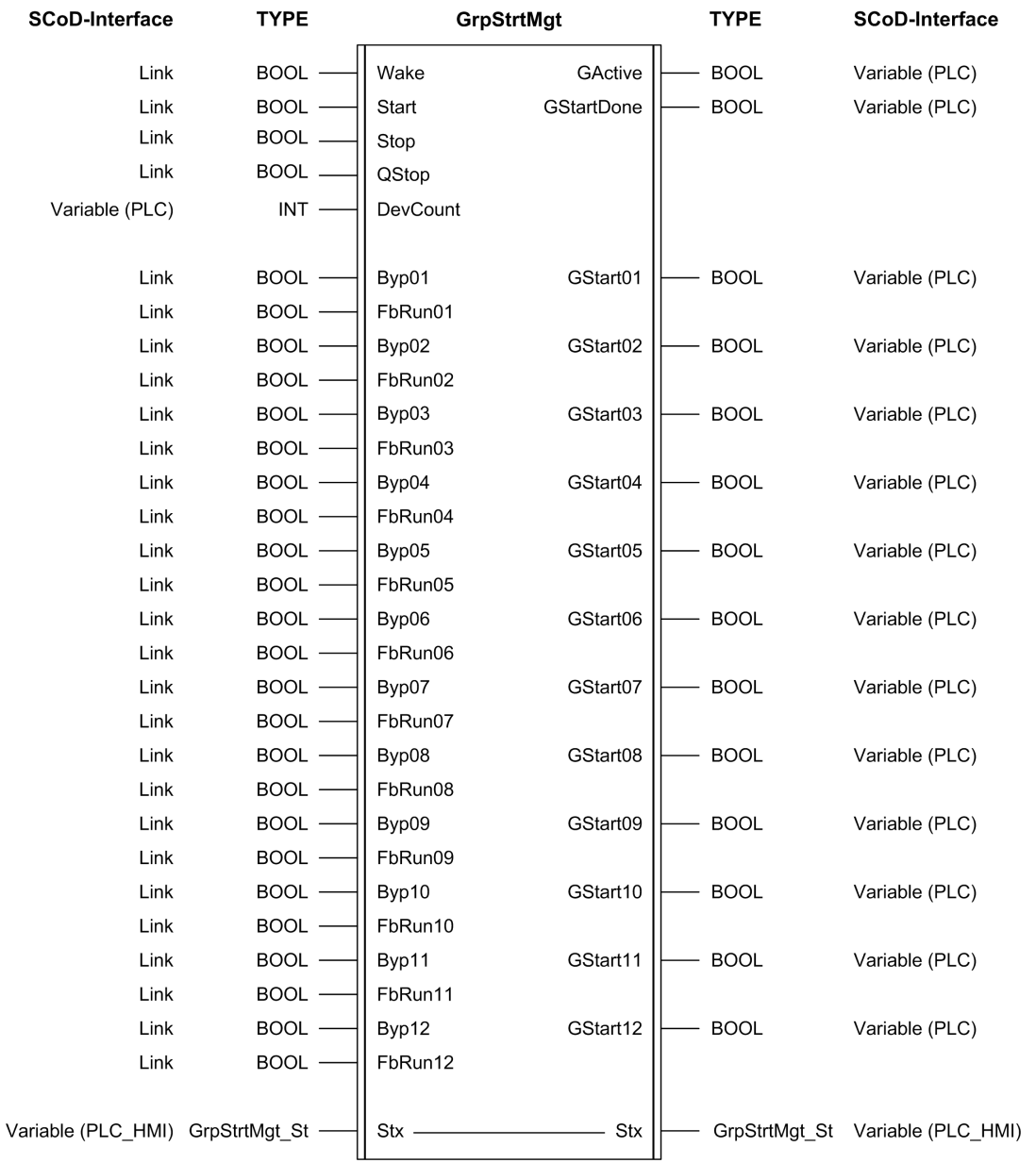
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the `Stx` variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type `HMI` is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the `GrpStrtMgt` block.



SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Display format (time)	Stx.TStart1	0	**Configuration**	1	Display format (time)
Device Count	DevCount	12 - Twelve	**Configuration**	1	Device count
Time delay to start device 01	Stx.TStart1	0	**Delay**	1	Time delay to start device 1
Time delay to start device 02	Stx.TStart2	0	**Delay**	7	Time delay to start device 2
Time delay to start device 03	Stx.TStart3	0	**Delay**	2	Time delay to start device 3
Time delay to start device 04	Stx.TStart4	0	**Delay**	8	Time delay to start device 4
Time delay to start device 05	Stx.TStart5	0	**Delay**	3	Time delay to start device 5
Time delay to start device 06	Stx.TStart6	0	**Delay**	9	Time delay to start device 6
Time delay to start device 07	Stx.TStart7	0	**Delay**	4	Time delay to start device 7
Time delay to start device 08	Stx.TStart8	0	**Delay**	10	Time delay to start device 8
Time delay to start device 09	Stx.TStart9	0	**Delay**	5	Time delay to start device 9
Time delay to start device 10	Stx.TStart10	0	**Delay**	11	Time delay to start device 10
Time delay to start device 11	Stx.TStart11	0	**Delay**	6	Time delay to start device 11
Time delay to start device 12	Stx.TStart5	0	**Delay**	12	Time delay to start device 12
Device 1 Name	Dev01Desc	Not available	**Device Names**	1	Device 1 name
Device 2 Name	Dev02Desc	Not available	**Device Names**	7	Device 2 name
Device 3 Name	Dev03Desc	Not available	**Device Names**	2	Device 3 name
Device 4 Name	Dev04Desc	Not available	**Device Names**	8	Device 4 name
Device 5 Name	Dev05Desc	Not available	**Device Names**	3	Device 5 name
Device 6 Name	Dev06Desc	Not available	**Device Names**	9	Device 6 name
Device 7 Name	Dev07Desc	Not available	**Device Names**	4	Device 7 name
Device 8 Name	Dev08Desc	Not available	**Device Names**	10	Device 8 name

Name	Variable	Initial Value	Property Tab	Position	Description
Device 9 Name	Dev09Desc	Not available	**Device Names**	5	Device 9 name
Device 10 Name	Dev10Desc	Not available	**Device Names**	11	Device 10 name
Device 11 Name	Dev11Desc	Not available	**Device Names**	6	Device 11 name
Device 12 Name	Dev12Desc	Not available	**Device Names**	12	Device 12 name

GrpStrtMgt - Representation in the HMI

Group Start Management (Genie)

The GrpStrtMgt block is represented in the HMI by the following genie:

Genie Elements

Symbol	Meaning
	Opens the GrpStrtMgt popup.
	The instance name of the GrpStrtMgt block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

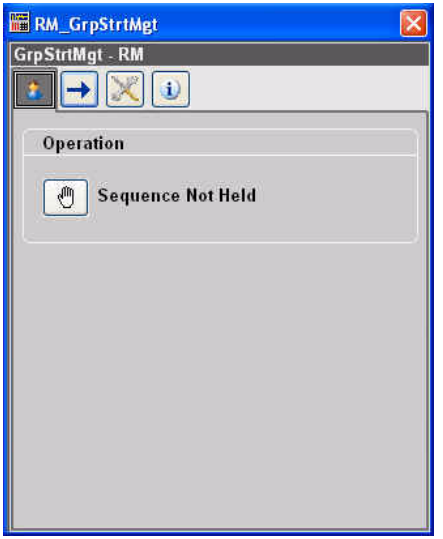
Symbol	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Popup

The GrpStrtMgt block popup has the following tabs:

- **Operator** tab
- **Extended** tab
- **Configuration** tab
- **Information** tab

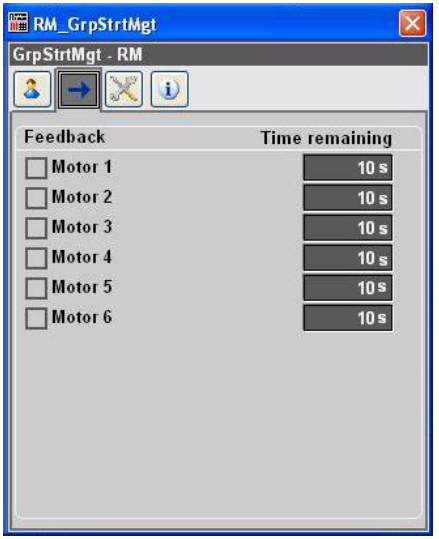
Operator Tab



The following table describes the sub-section **Operation**:

Item	Description
Sequence Not Held	Holds the sequence of a control module.

Extended Tab

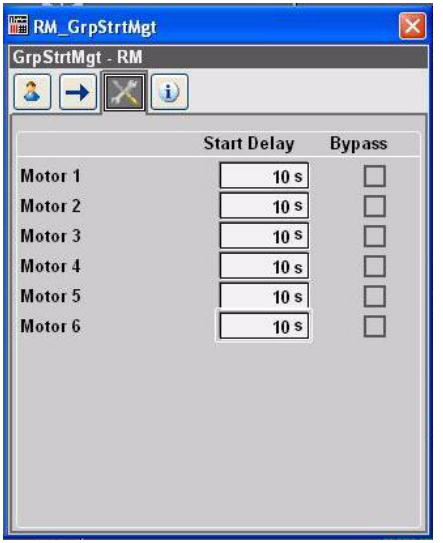


The following table describes the sub-section **Feedback**:

Item	Animation	Description
Feedback	Steady green	Device started feedback is obtained.
	Steady gray	Device started feedback is not obtained.
Time remaining	Not available	Displays the time remaining to the device x (x: 01...12).

This tab shows only the devices as per the device count (`DevCount`) configured.

Configuration Tab



The following table describes the sub-section **Configuration**:

Item	Animation	Description
Start Delay	Not available	Displays the start delay for device x (x: 1...12), configurable from HMI.
Bypass	Steady green	Timer bypass feedback for device x (x: 1…12) is obtained.
	Steady gray	Timer bypass feedback for device x (x: 1…12) is not obtained.

This tab shows only the devices as per the device count (`DevCount`) configured.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

GrpStrtMgt_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type GrpStrtMgt_St for the GrpStrtMgt block:

Element	Type	Description
TStartx (x: 01...12)	INT	Delay to start devices x (x: 01...12)
TRStartx (x: 01...12)	INT	Time remaining to start devices x (x: 01...12)
DevCount	INT	Device count
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Stw2	WORD	Status word 2 to interact with HMI

Bit Description

Bit description for the Ctw element:

Bit	Description
0	Hold
1...13	Not used
14	Alarm mask
15	Alert mask

Bit description for the Stw1 element:

Bit	Description
0...11	Feed back run x (x: 01...12)
12...15	Not used

Bit description for the Stw2 element:

Bit	Description
0...11	Bypass of timer x (x: 01...12)
12	Group active
13	Group start done
14...15	Not used

9.3 GrpStpMgt: Group Stop Management

Overview

This section describes the GrpStpMgt block.

What's in this Section?

This section contains the following topics:

Topic	Page
GrpStpMgt - Description	372
GrpStpMgt - Representation as SCoD in UAG	376
GrpStpMgt - Representation in the HMI	380
GrpStpMgt_St Derived Data Type	384

GrpStpMgt - Description

General

The GrpStpMgt block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB GrpStpMgt
Vijeo Citect	Genie GrpStpMgt
UAG	SCoD GrpStpMgt

Function Description

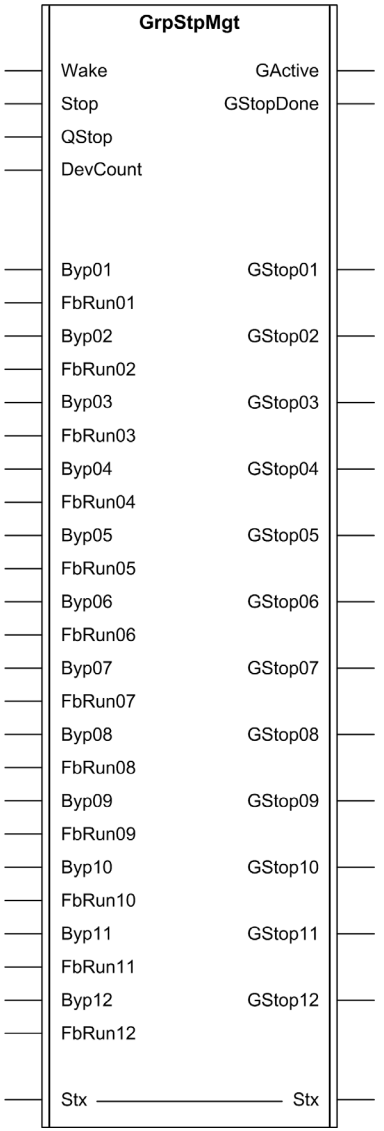
The GrpStpMgt block is used to manage the sequential stopping of the devices in the group.

This block is used as an intermediate block between the GrpCtrl and device (motors/actuators), and can handle up to 12 devices. If there are more than 12 devices, the GrpStpMgt can be extended by connecting the GStopDone output of this block to the Stop pin of the next GrpStpMgt block.

The stop status of every device is available through the feedback device input FbRunx (x: 1...12). Time delay to stop the next device is configured from the popup through TStopx (x: 1...12). The popup also shows the remaining time to stop the next device TRStopx (x: 1...12). This configured time is bypassed and an immediate **Stop** command is provided by assigning specific conditions to the input Bypx (x: 1...12). After the elapsed time equals the configured time delay TStopx, the output GStopx (x: 1...12) will give an output. The GStopx output is connected to the Stop pin of the device/block. Hence the device/block gets the **Stop** command from the GrpStpMgt block.

Representation in FBD

The following figure shows the FBD representation of the `GrpStpMgt` block.



Parameter Description

The following table describes the input parameters of the GrpStpMgt block:

Parameter	Type	Description	Details
Wake	BOOL	Group wake-up	Group wake-up and request for group availability check. • Wake = 1: Group is active. • Wake = 0: Group is in Sleep mode. This input is connected from output GWake of GrpCtrl.
Start	BOOL	**Start** command	**Start** command starts all the devices in the group sequentially.
Stop	BOOL	**Stop** command	Stop = 0...1: The block initiates the **Stop** command and GActive = 1.
QStop	BOOL	Quick stop command (**QStop)**	QStop = 0...1: The block deactivates the **Stop** command and GActive = 0.
DevCount	INT	Device count	The number of devices are present in the group.
Bypx (x: 01...12)	BOOL	Bypass of timer Tx (x: 01...12)	Bypx (x: 01...12) = 1: Bypass the timer and stop the device x (x: 01...12) immediately.
FbRunx (x: 01...12)	BOOL	Run feedback device x (x: 01...12)	Device x (x: 01...12) are not active or stopped.

The following table describes the output parameters of the GrpStpMgt block:

Parameter	Type	Description	Details
GActive	BOOL	Group active	The GrpStpMgt is stopped in the group and is active in the group.
GStopDone	BOOL	Group stop activity is done	This output is set when all the devices in the group are stopped. **NOTE:** It is connected to the next GrpStpMgt block.
GStopx (x: 01...12)	BOOL	Group stop command (**GStop)** x (x: 01...12)	GStopx (x: 01...12) = 1: Stops device x (x: 01...12). **NOTE:** To be connected to input Stop of the device x (x: 01...12).

The following table describes the input/output parameter of the `GrpStpMgt` block:

Parameter	Type	Description
`Stx`	`GrpStpMgt_St`	Status and control of the `GrpStpMgt` block. Refer to the Structure to interface with the HMI *(see page 384)*.

GrpStpMgt - Representation as SCoD in UAG

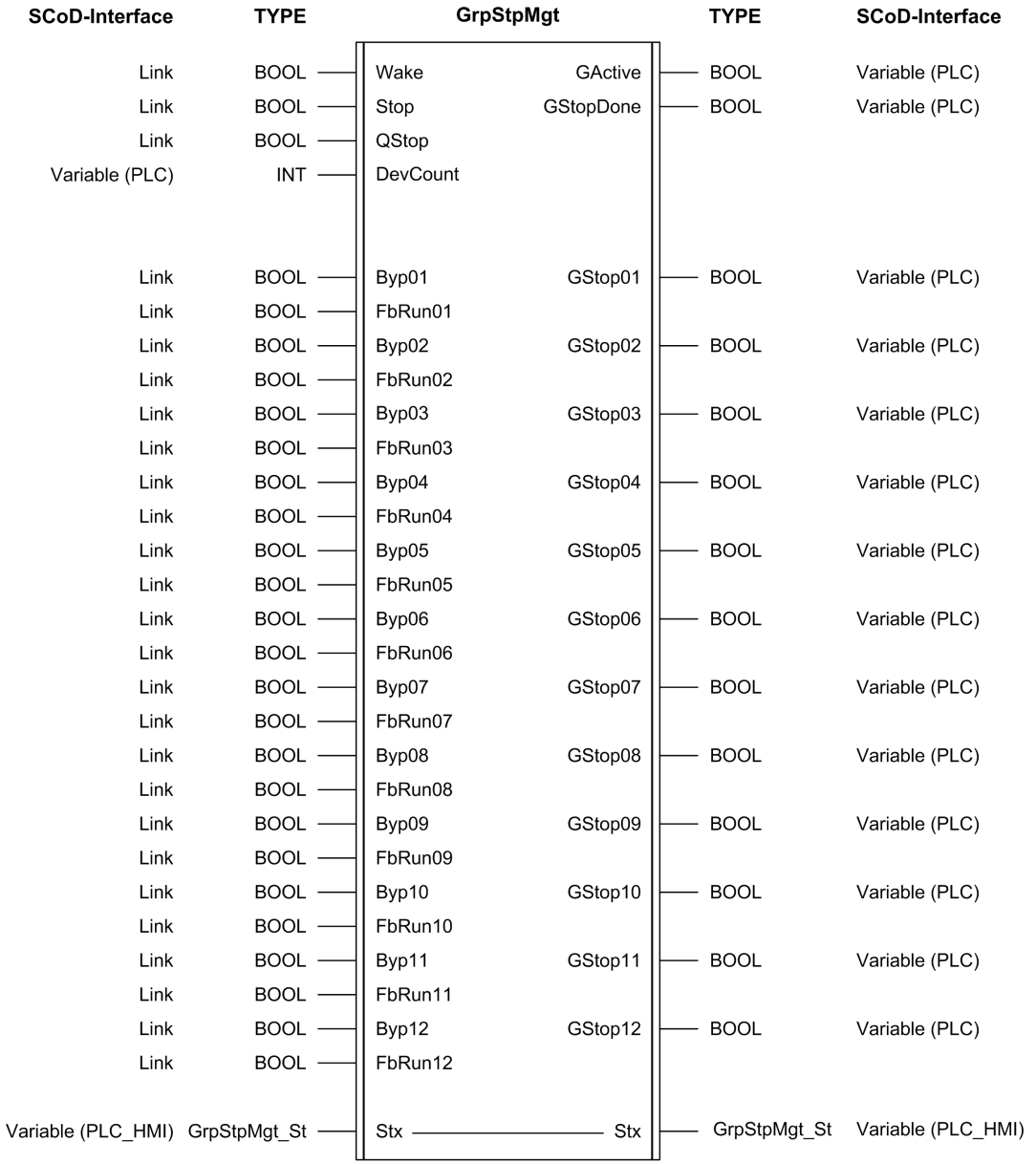
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the `Stx` variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type `HMI` is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the `GrpStpMgt` block.



SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Device Count	DevCount	12 - Twelve	**Configuration**	1	Device count
Display format (time)	Stx.TStop1	0	**Configuration**	7	Display format (time)
Time delay to stop device 01	Stx.TStop1	0	**Delay**	1	Time delay to stop device 1
Time delay to stop device 02	Stx.TStop2	0	**Delay**	7	Time delay to stop device 2
Time delay to stop device 03	Stx.TStop3	0	**Delay**	2	Time delay to stop device 3
Time delay to stop device 04	Stx.TStop4	0	**Delay**	8	Time delay to stop device 4
Time delay to stop device 05	Stx.TStop5	0	**Delay**	3	Time delay to stop device 5
Time delay to stop device 06	Stx.TStop6	0	**Delay**	9	Time delay to stop device 6
Time delay to stop device 07	Stx.TStop7	0	**Delay**	4	Time delay to stop device 7
Time delay to stop device 08	Stx.TStop8	0	**Delay**	10	Time delay to stop device 8
Time delay to stop device 09	Stx.TStop9	0	**Delay**	5	Time delay to stop device 9
Time delay to stop device 10	Stx.TStop10	0	**Delay**	11	Time delay to stop device 10
Time delay to stop device 11	Stx.TStop11	0	**Delay**	6	Time delay to stop device 11
Time delay to stop device 12	Stx.TStop12	0	**Delay**	12	Time delay to stop device 12
Device 1 Name	Dev01Desc	Not available	**Device Names**	1	Device 1 name
Device 2 Name	Dev02Desc	Not available	**Device Names**	7	Device 2 name
Device 3 Name	Dev03Desc	Not available	**Device Names**	2	Device 3 name
Device 4 Name	Dev04Desc	Not available	**Device Names**	8	Device 4 name
Device 5 Name	Dev05Desc	Not available	**Device Names**	3	Device 5 name
Device 6 Name	Dev06Desc	Not available	**Device Names**	9	Device 6 name
Device 7 Name	Dev07Desc	Not available	**Device Names**	4	Device 7 name
Device 8 Name	Dev08Desc	Not available	**Device Names**	10	Device 8 name

Name	Variable	Initial Value	Property Tab	Position	Description
Device 9 Name	Dev09Desc	Not available	**Device Names**	5	Device 9 name
Device 10 Name	Dev10Desc	Not available	**Device Names**	11	Device 10 name
Device 11 Name	Dev11Desc	Not available	**Device Names**	6	Device 11 name
Device 12 Name	Dev12Desc	Not available	**Device Names**	12	Device 12 name

GrpStpMgt - Representation in the HMI

Group Stop Management (Genie)

The GrpStpMgt block is represented in the HMI by the following genie.

Genie Elements

Symbol	Meaning
	Opens the GrpStpMgt popup.
	The instance name of the GrpStpMgt block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

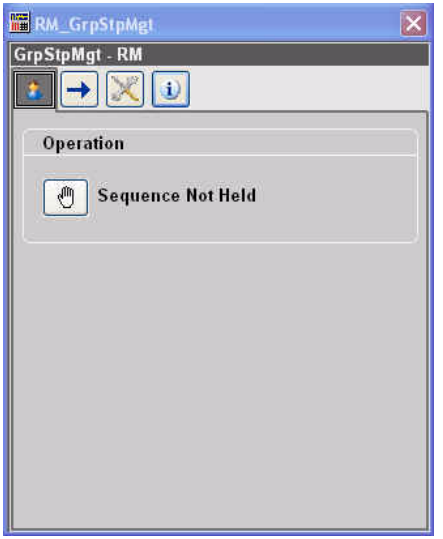
Symbol	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Popup

The GrpStpMgt block popup has the following tabs:

- **Operator** tab
- **Extended** tab
- **Configuration** tab
- **Information** tab

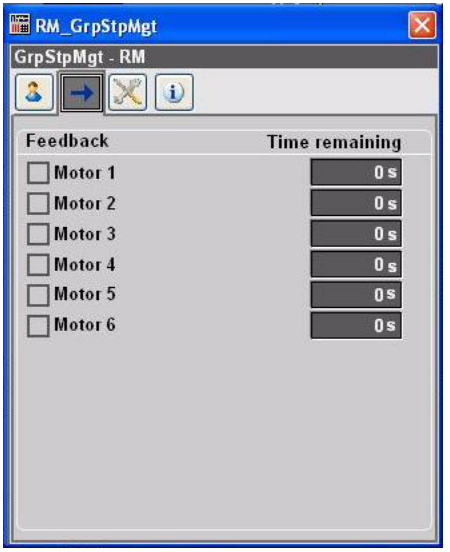
Operator Tab



The following table describes the sub-section **Operation**:

Item	Description
Hold Sequence	Holds the sequence of a control module.

Extended Tab

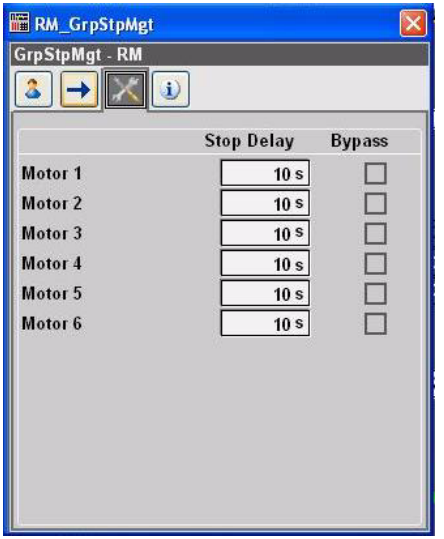


The following table describes the sub-section **Feedback**:

Item	Animation	Description
Feedback	Steady green	Device stopped feedback is obtained.
	Steady gray	Device stopped feedback is not obtained.
Time remaining	Not available	Displays the time remaining to the stop for device x (x: 01...12).

This tab shows only the devices as per the device count (`DevCount`) configured.

Configuration Tab



The following table describes the sub-section **Configuration**:

Item	Animation	Description
Stop Delay	Not available	Displays the stop delay for device x (x: 01...12), configurable from HMI.
Bypass	Steady green	Timer bypass feedback for device x (x: 01...12) is obtained.
	Steady gray	Timer bypass feedback for device x (x: 01...12) is not obtained.

This tab shows only the devices as per the device count (`DevCount`) configured.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

GrpStpMgt_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type GrpStpMgt_St for the GrpStpMgt block:

Element	Type	Description
TStopx (x: 01...12)	INT	Time delay to stop device x (x: 01...12)
TRStopx (x: 01...12)	INT	Time remaining to stop device x (x: 01...12)
DevCount	INT	Device count
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Stw2	WORD	Status word 2 to interact with HMI

Bit Description

Bit description for the Ctw element:

Bit	Description
0	Holds the group stop command (**GStop)**
1...13	Not used
14	Alarm mask
15	Alert mask

Bit description for the Stw1 element:

Bit	Description
0...11	Stop run feedback device x (x: 01...12)
12...15	Not used

Bit description for the Stw2 element:

Bit	Description
0...11	Bypass of timer x (x: 01...12)
12	Group active
13	Group stop done
14...15	Not used

9.4 BCnvyr1d: Belt Conveyor 1 Direction

Overview

This section describes the BCnvyr1d block.

What's in this Section?

This section contains the following topics:

Topic	Page
BCnvyr1d - Description	386
BCnvyr1d - Representation as SCoD in UAG	400
BCnvyr1d - Representation in the HMI	403
BCnvyr1d_St Derived Data Type	414

BCnvyr1d - Description

General

The BCnvyr1d is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB BCnvyr1d
Vijeo Citect	Genie BCnvyr1d
UAG	SCoD BCnvyr1d

Function Description

This BCnvyr1d is a customised 1 direction 1 speed motor block specifically for belt conveyors that includes the inoperable conditions like belt sway, pull chord inoperable. In addition, it also checks the process feedback, such as **Zero Speed** switch, belt running empty, and so on.

This block supports 3 modes of operation:
- Auto mode
- Manual mode
- Local mode

By default, the conveyor will be in Auto mode. The conveyor is forced to Auto or Local mode if the mode selection input pin (AutoSel or LocalSel) is HIGH. Local pin always have the highest priority. When the Local mode is selected, the block functions in Local mode irrespective of other mode selections.

If the mode selection input pin (AutoSel or LocalSel) is LOW, operator changes the mode of the conveyor from the popup of the conveyor block. When the conveyor is under maintenance, the Out of Service mode can be activated from the popup.

Auto Mode Operation

A wake-up command (**Wake**) is given by the GrpCtrl block to wake-up the motor from Sleep mode. A group start-up pulse command (**Start**) is given either from the GrpCtrl or GrpStrtMgt or RouteMgt or from previous device in the group. This sets the output GActive as HIGH and the motor starts in the selected direction.

Ready output (Ready) is set HIGH when all interlocks and conditions are HIGH. Interlock output (ILck) is set HIGH when inputs starting interlock (StartILck), safety interlock (SftyILck), sequential interlock (SeqILck) and process interlock (ProcILck) are set HIGH.

After the start command (**Start**) is issued and output (OP) is set HIGH within the specified run feedback delay, the electrical run feedback (FbRun) has to be obtained. This electrical run feedback is continuously monitored by the motor block. If it is not operational, motor is tripped, and an alarm is generated.

After the start command (**Start**) is issued and motor is operational. If run feedback (FbRun) is not available for 500 ms, the motor is tripped, and an alarm is generated.

After the start command (**Start**) is issued and the output (OP) is set HIGH within the specified process feedback delay, the process feedback (FbProc) has to be obtained. This process feedback is continuously monitored by the motor block. If it is not operational, motor is tripped, and an alarm is generated.

After the start command (**Start**) is issued, output (OP) is set HIGH, and motor is operational. If process feedback (FbProc) is not available for specified loss of process feedback delay, the motor is tripped, and an alarm is generated.

NOTE: The process feedback can be bypassed by selecting the **Bypass Process f/b** button from the HMI.

NOTE: Group Active, Sequential Start and Sequential Stop are applicable in Auto mode only.

The start command **(Start)** is issued, and output (OP) is set HIGH when the run feedback (FbRun) and process feedback (FbProc) are obtained. Run output (Run) is set HIGH, and pulse command (**SeqStart**) is given to the next device to start.

Group stop command (**Stop**) or group quick stop command (**QStop**) sets the output command (**OP**) as LOW. Group stop command (**Stop**) is received by the motor from the GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group. Group quick stop command (**GQStop**) is received by the motor from the GrpCtrl block.

The **Stop** command is issued and output (OP) is set LOW when the run feedback (FbRun) and process feedback (FbProc) are set LOW. Run output (**Run**) is set LOW and command (**SeqStop**) is given to the next device to stop.

The quick stop command (**QStop**) is issued and output (OP) is set LOW and sequential stop command (**SeqStop**) is given to the next device to stop.

Manual Mode Operation

Run and **Stop** commands are issued by the operator from the popup of the conveyor block. Run and process feedback monitoring are applicable as in Auto mode.

NOTE: Sequential interlock is not applicable in Manual mode

Local Mode Operation

Stage	Description
1	Start-up pulse command (**LStart**) comes from the local control panel of the motor that sets the output `OP` HIGH to start the conveyor in the selected direction.
2	Stop command (**LStop**) also comes from the local control panel of the conveyor. This input must be HIGH for the conveyor to be started and is applicable in all the operation modes. **NOTE:** This serves as an Emergency Stop (**EStop**) command.

Run and process feedback monitoring are applicable as in Auto mode.

Interlock Types

There are 4 types of interlocks. The result of individual interlock condition blocks is connected with the following interlock pins:

Interlock Name	Description
Starting Interlock	These interlocks are valid before conveyor start-up only.
Safety Interlock	All the safety conditions are taken into consideration. This input is active in all the modes.
Sequential Interlock	All the sequential conditions are taken into consideration. This input is active only in Auto mode.
Process Interlock	All the process conditions are taken into consideration. This input is active in Auto and Manual mode.

Feedback

The status of the interlock, for example, active or not active is displayed in the popup. Detailed description of the interlocks is displayed in the **Interlock** tab of the popup.

The availability (`Avail`) of the motor is continuously monitored. If it is not operational, the motor is tripped and an alarm is generated.

The overload feedback (`OverLoad`) of the motor is continuously monitored. If it is not operational, the motor is tripped and an alarm is generated.

An emergency stop signal (`EStop`) comes from an emergency shut-down push button and/or from a pull rope if it is used for belt conveyors.

The in-built safety switches and monitoring conditions for the conveyor is listed as follows:

- **Belt Sway** switch feedback is continuously monitored. If the conveyor is running and the occurrence of the belt sway conveyor is stopped, then an alarm is generated after a delay.
- **Belt Empty** switch feedback is continuously monitored. If the conveyor is running with no material, then the conveyor is stopped, and an alarm is generated after a delay.
- **Pull Cord** switch feedback is continuously monitored. If a **Pull Cord** switch has been pulled, then the conveyor is stopped, and an alarm is generated after a delay. If a modbus link is provided in the architecture to get the information of the pull cord number which was pulled, then this information can be viewed in the popup of the conveyor.
- **Belt Tension** switch feedback is continuously monitored. If the belt tension is in inoperable condition, then the conveyor is stopped and an alarm is generated after a delay.
- **Belt Chute** switch feedback is continuously monitored. If the chute gets jammed, the conveyor is stopped, and an alarm is generated after a delay.

These feedbacks are shown in the **Extended** tab of the conveyor 1 direction popup.

The alerts and trips are indicated in the SCADA. The number of active alarms are shown in the popup of the conveyor. The details of the active alarms are shown in the **Alarm** tab of the popup. Based on the user configured conditions, the alarms for this block can be inhibited by connecting the logic to the input pin `AlmInh`. The trip alarms can be reset by the input pin `AlmRst`, which is connected from `GrpCtrl` block or by the operator from popup.

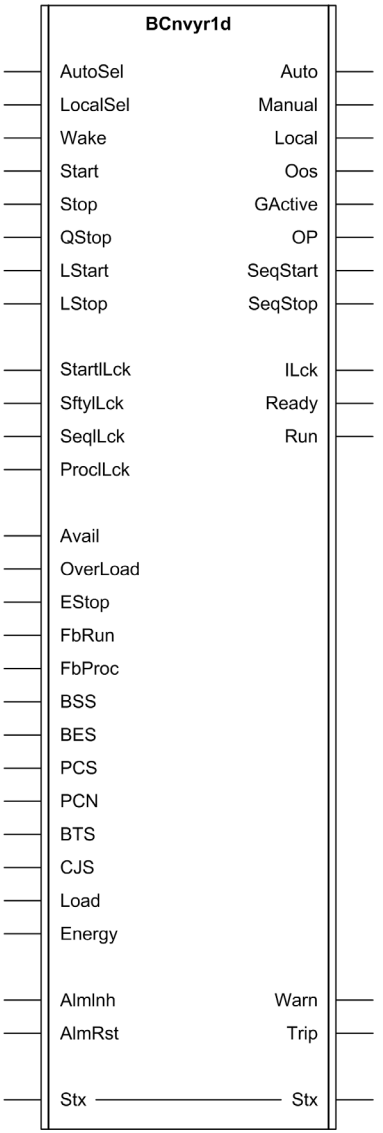
Energy and Maintenance

The load and energy values are inputs to the motor block; these values are displayed in the popup of the motor block. The run hours of the motor is displayed in the **Configuration** tab of the popup. After the maintenance of the motor, the run hours can be reset from the popup.

NOTE: The parameters run feedback delay, process feedback delay, loss of process feedback delay, inversion delay, and run hours reset have the user restrictions from HMI.

Representation in FBD

The following figure shows the FBD representation of the `BCnvyr1d` block.



Parameter Description

The following table describes the input parameter of the BCnvyr1d block:

Parameter	Type	Description	Details
AutoSel	BOOL	Auto mode selection	Selection of operation mode (Auto mode): • AutoSel = 1: Block is forced to Auto mode. Mode selection option in HMI is grayed out. • AutoSel = 0: Block is set free for operator to select the mode from HMI.
LocalSel	BOOL	Local mode selection	Selection of operation mode (Local mode): • LocalSel = 1: Block is forced to Local mode. Mode selection option in HMI is grayed out. • LocalSel = 0: Block is set free for operator to select the mode from HMI.
Wake	BOOL	Group wake-up	Group wake-up and request for group availability check: • Wake = 1: Group is active. • Wake = 0: Group is in Sleep mode. This input is connected from output GWake of GrpCtrl.
Start	BOOL	Group start	**Start** command to the motor. • Start = 1: OP pin is set to 1, this is applicable in Auto mode only. This input is connected from the GStart pin of GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group.
Stop	BOOL	Group stop	**Stop** command to the motor. • Stop = 1: OP pin is not set to 1, this is applicable in Auto mode only. This input is connected from the GStop pin of GrpCtrl or GrpStpMgt or RouteMgt or previous device in the group.
QStop	BOOL	Group quick stop	**Stop** command to the motor. • QStop = 1: OP pin is set to 0, this is applicable in Auto mode only. This input is connected from the GQStop pin of GrpCtrl.

Parameter	Type	Description	Details
LStart	BOOL	Local start	**Start** command to the motor. • LStart = 1: OP pin is set to 0, this is applicable in Local mode only. This input is connected from the **Start** push button of local IO panel.
LStop	BOOL	Local stop	**Stop** command to the motor. • LStop = 1: OP pin is set to 0, this is applicable in Local mode only. This input is connected from the **Stop** push button of local IO panel.
StartILck	BOOL	Starting interlock	• StartILck = 1: All the starting interlock conditions are operational. • StartILck = 0: All the starting interlock conditions are not operational. Applicable only in Auto and Manual modes of operation. Applicable only before starting of the motor. After motor is started, StartILck is not considered.
SftyILck	BOOL	Safety interlock	• SftyILck = 1: The safety interlock conditions are operational. • SftyILck = 0: The safety interlock conditions are not operational, trips the motor and generates an alarm. Applicable in all the modes of operation (Auto, Manual, and Local).
SeqILck	BOOL	Sequential interlock	• SeqILck = 1: All the sequential interlock conditions are operational. • SeqILck = 0: All the sequential interlock conditions are not operational, trips the device and generates an alarm. Applicable in Auto mode only.

Parameter	Type	Description	Details
ProcILck	BOOL	Process interlock	• ProcILck = 1: All the process interlock conditions are operational. • ProcILck = 0: All the process interlock conditions are not operational, trips the device and generates an alarm. Applicable in Auto and Manual operation modes only.
Avail	BOOL	Availability of motor	• Avail = 1: Motor is electrically available. • Avail = 0: Motor is electrically not available. This signal comes from the MCC outgoing circuit (electrical protection active, circuit breaker switched off, drawer out, and so on).
OverLoad	BOOL	Overload	• OverLoad = 1: Motor is not in overload condition. • OverLoad = 0: Motor is in overload condition. **Stop** command to the motor, OP = 0. This signal comes from the MCC outgoing circuit (electrical protection active).
EStop	BOOL	Emergency stop	• EStop = 1: Plant process condition is operational for the motor to run. • EStop = 0: Emergency situation. Plant process condition is not operational for the motor to run. **Stop** command to the motor, OP = 0. Applicable in all operation modes of operation (Auto, Manual, and Local). This input is connected from the **Emergency Stop** push button in the field.

Parameter	Type	Description	Details
FbRun	BOOL	Run feedback of motor - Electrical	Once the motor is given a **Start** command from output OP, its electrical run feedback is continuously monitored. This signal comes from the MCC outgoing circuit. • OP = 1 and FbRun = 0, an alarm is generated. • OP = 0 and FbRun = 1, an alarm is generated.
FbProc	BOOL	Run feedback of motor - Process	Once the motor is given a **Start** command from output OP, its electrical run feedback is continuously monitored. This signal comes from speed monitoring devices, **Zero Speed** switch, proximity switches, and so on. • OP = 1 and FbProc = 0, **Stop** command is given to the motor. OP = 0 is given, an alarm is generated after time delay. • OP = 0 and FbProc = 1, an alarm is generated after time delay.
BSS	BOOL	**Belt Sway** switch	• BSS = 1: Conveyor condition is operational to run. • BSS = 0: Belt sway is occurred. Conveyor is not operational to run. OP = 1 and BSS = 0, **Stop** command is given to the motor. OP = 0: An alarm is generated. Applicable in all operation modes (Auto, Manual and Local). This input is connected from the **Belt Sway** switch mounted near the belt in field.

Parameter	Type	Description	Details
BES	BOOL	**Belt Empty Sensor** switch	• BES = 1: Conveyor condition is operational to run. • BES = 0: Conveyor is running at empty with no material on it. Conveyor condition is not operational to run. OP = 1 and BES = 0, **Stop** command is given to the motor. OP = 0: An alarm is generated. Applicable in all operation modes (Auto, Manual and Local). This input is connected from the belt empty sensor mounted near the belt in field.
PCS	BOOL	**Pull Cord** switch	• PCS = 1: Conveyor condition is operational to run. • PCS = 0: A **Pull Cord** switch is pulled by somebody or remains inoperable. Conveyor condition is not operational to run. OP = 1 and PCS = 0, **Stop** command is given to the motor. OP = 0: An alarm is generated. Applicable in all operation modes (Auto, Manual and Local). This input is connected from the **Pull Cord** switches sequentially and mounted near the belt in field.
PCN	INT	**Pull Cord** switch number	This gives information on which pull cord is exactly inoperable.
BTS	BOOL	**Belt Tension** switch	• BTS = 1: Conveyor condition is operational to run. • BTS = 0: Belt tension is not ok. Conveyor condition is not operational to run. OP = 1 and BTS = 0, **Stop** command is given to the motor. OP = 0: An alarm is generated. Applicable in all operation modes (Auto, Manual and Local). This input is connected from the **Belt Tension Ok** switch and mounted near the belt in field.

Parameter	Type	Description	Details
CJS	BOOL	**Chute Jam** switch	• CJS = 1: Conveyor condition is operational to run. • CJS = 0: Belt chute is jammed. Conveyor condition is not operational to run. OP = 1 and CJS = 0, **Stop** command is given to the motor. OP = 0: An alarm is generated. Applicable in all operation modes (Auto, Manual and Local). This input is connected from the **Belt Chute Jam** switch and mounted near the belt in field.
Load	REAL	Load value of the motor	Load value of the motor is in A/kW.
Energy	REAL	Energy value of the motor	Energy value of the motor is in kWh.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

The AlmInh parameter to inhibit alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the BCnvyr1d block:

Parameter	Type	Description	Details
AlmInh	BOOL	Alarm inhibition	• AlmInh = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • AlmInh = 0: Alarms are not inhibited.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

To reset, alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameters of the BCnvyr1d block:

Parameter	Type	Description	Details
AlmRst	BOOL	Reset	• AlmRst = 1: Resets all the alarms generated by the block. • AlmRst = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from GRst pin of GrpCtrl block and resets the trip alarm.

The following table describes the output parameters of the BCnvyr1d block:

Parameter	Type	Description	Details
Auto	BOOL	Auto mode	• Auto = 1: Auto mode is selected. • Auto = 0: Auto mode is not selected. Motor operation starts in the group based on GStart and GStop.
Manual	BOOL	Manual mode	• Manual = 1: Manual mode is selected. • Manual = 0: Manual mode is not selected. Motor operation is based on commands given by the operator from HMI.
Local	BOOL	Local mode	• Local = 1: Local mode is selected. • Manual = 0: Local mode is not selected. Motor operation is based on commands **LStart** and **LStop** given from field.
Oos	BOOL	Out of Service	• Oos = 1: Out of Service mode is selected. • Oos = 0: Out of Service mode is not selected.
GActive	BOOL	Group active	The motor is started in the group and is active in the group. • IfOP, FbRun and FbProc = 1, then GActive = 1. If OP, FbRun and FbProc = 1, then GActive = 1.

Parameter	Type	Description	Details
OP	BOOL	Output command (**OP**) to motor	• OP = 1: Motor is giving command to stop. • OP = 0: Motor is giving command to run. The output switches on the contactor or circuit breaker is controls the motor in forward direction.
SeqStart	BOOL	**Start** command to next motor in the group	• OP = 1: f GRun = 1, **Start** command (pulse output: SeqStart = 0...1) is given to the next motor connected in the group.
SeqStop	BOOL	**Stop** command to next motor in the group	• If SeqStop = 1, **Stop** command SeqStop = 1) is given to the next motor connected in the group. • SeqStop = 0:
ILck	BOOL	Interlock status	• ILck = 1: All the interlocks are operational. • ILck = 0: All the interlocks are not operational. There is an active interlock present.
Ready	BOOL	Motor ready to start	• Ready = 1: All the conditions to start the motor are operational. • Ready = 0: All the conditions to start the motor are not operational. Motor will stop giving **Start** command.
Run	BOOL	Motor running	Motor is running. This output is given if the motor is running in forward direction, and run forward feedback is received. Applicable in all the operation modes.
Warn	BOOL	Alert	• Warn = 1: The motor is operational with no active alert present. • Warn = 0: The motor is operational with an active alert present.
Trip	BOOL	Trip	• Trip = 1: The motor is operational with no active trip alarm present. • Trip = 0: The motor is tripped with an active trip alarm present.

The following table describes the input/output parameter of the `BCnvyr1d` block:

Parameter	Type	Description
`Stx`	`BCnvyr1d_St`	Status and control of belt conveyor 1 direction. Refer to the Structure to interface with the HMI *(see page 414)*.

`BCnvyr1d` - Representation as SCoD in UAG

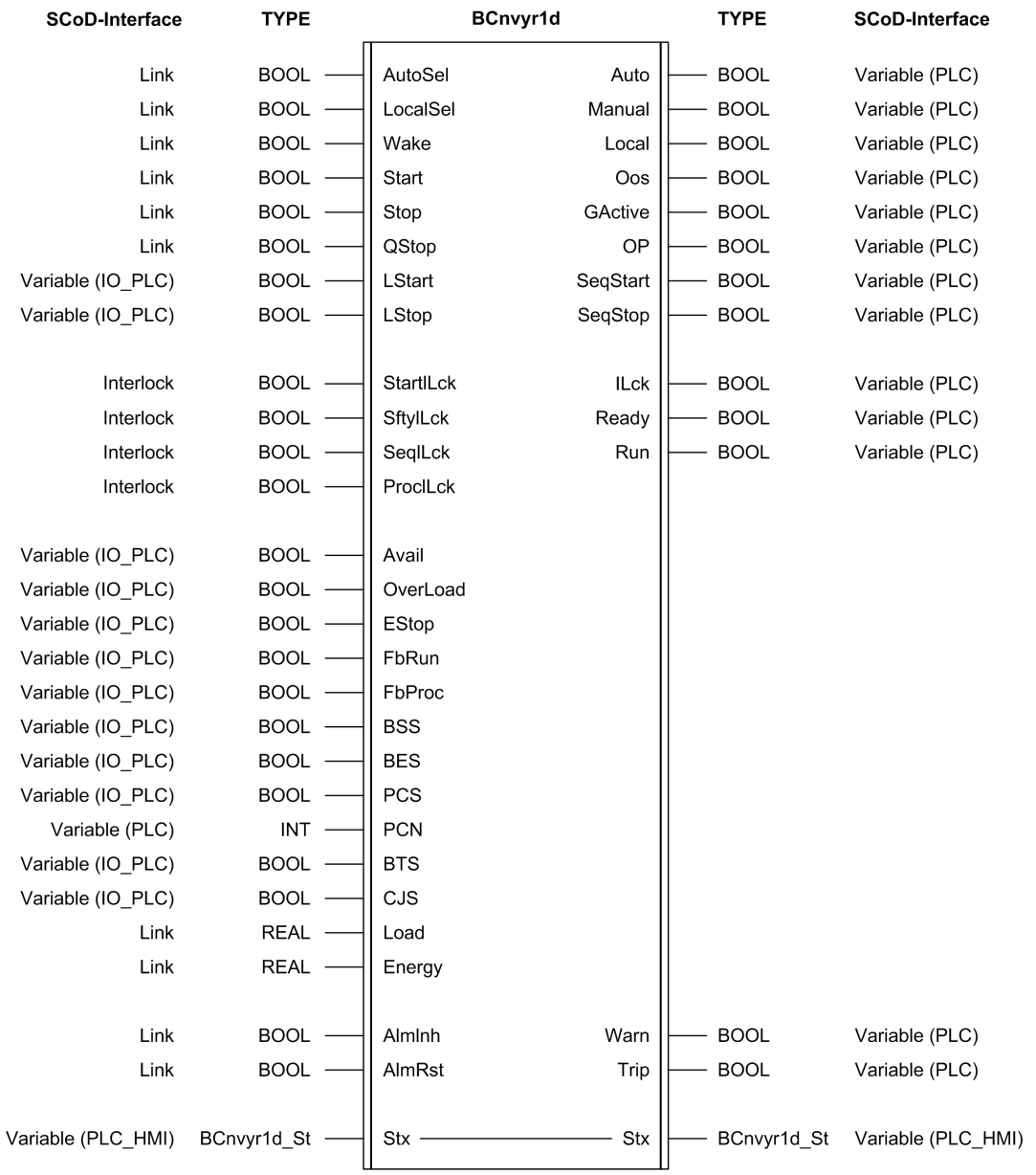
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the `Stx` variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type `HMI` is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the `BCnvyr1d` block.



SCoD Properties

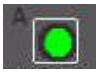
Name	Variable	Initial Value	Property Tab	Position	Description
Display Format (time)	`Stx.RunHr`	Not available	**Configuration**	1	Run hours of motor
Run fb delay (sec)	`Stx.TRunFb`	0	**Configuration**	2	Run feedback delay
Loss of process feedback delay	`Stx.TLProcFb`	0	**Configuration**	3	Loss of process feedback delay
Load measurement Unit	`Stx.Load`	Not available	**Configuration**	4	Load of motor (A/kW)
Display format (SP)	`Stx.Load`	Not available	**Configuration**	7	Load of motor (A/kW)
Process feedback delay	`Stx.TProcFb`	0	**Configuration**	8	Process feedback delay
Energy measurement Unit	`Stx.Energy`	Not available	**Configuration**	10	Energy of motor (kWh)
Load Eng. Min	`Stx.Load`	Not available	**Range**	1	Sets the minimum load Eng. value
Energy Eng. Min	`Stx.Energy`	Not available	**Range**	2	Sets the minimum energy Eng. value
Load Eng. Max	`Stx.Load`	Not available	**Range**	7	Sets the maximum load Eng. value
Energy Eng. Max	`Stx.Energy`	Not available	**Range**	8	Sets the maximum energy Eng. value
Area Information	`AreaInfo`	Not available	**Device Data**	1	Area information
Asset ID	`AssetID`	Not available	**Device Data**	7	Asset ID

BCnvyr1d - Representation in the HMI

Belt Conveyor 1 Direction (Genie)

The BCnvyr1d block is represented in the HMI by the following genie.

Genie Elements

Symbol	Meaning
	Opens the BCnvyr1d popup.
	The instance name of the BCnvyr1d block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

Item	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Animation Description of the Genie

Symbol	Animation	Meaning
	Line color: Steady light gray Fill color: Steady light gray	Belt conveyor is in Sleep mode.
	Line color: Steady dark gray Fill color: Steady white	• Belt conveyor is awake or in Local mode or Manual mode and ready. • Belt conveyor is awake or in Local mode or Manual mode and ready after alarms are reset.
	Line color: Steady yellow Fill color: Steady white	Belt conveyor is awake and not ready.
	Line color: Steady dark gray Fill color: Flashing green	Belt conveyor is started and feedback is yet to be obtained or motor is stopping.

Symbol	Animation	Meaning
	Line color: Steady dark gray Fill color: Steady green	Belt conveyor is running.
	Line color: Flashing yellow Fill color: Steady green	Belt conveyor is running and an alert alarm is active.
	Line color: Steady yellow Fill color: Steady green	Belt conveyor is running and an alert alarm is acknowledged.
	Line color: Steady dark gray Fill color: Flashing red	Belt conveyor is tripped and a trip alarm is active.
	Line color: Steady dark gray Fill color: Steady red	Belt conveyor is tripped and a trip alarm is acknowledged.

Modes of operations

On the top left corner of the Belt conveyor 1 direction genie, the mode of operation is indicated.

Mode	Meaning
A	Belt conveyor is operating in the Auto mode.
M	Belt conveyor is operating in the Manual mode.
L	Belt conveyor is operating in the Local mode.
O	Belt conveyor is operating in the Out of Service mode.

Alarm Masking

Symbol	Meaning
	Trip or alert alarms in the Belt conveyor 1 direction are masked.

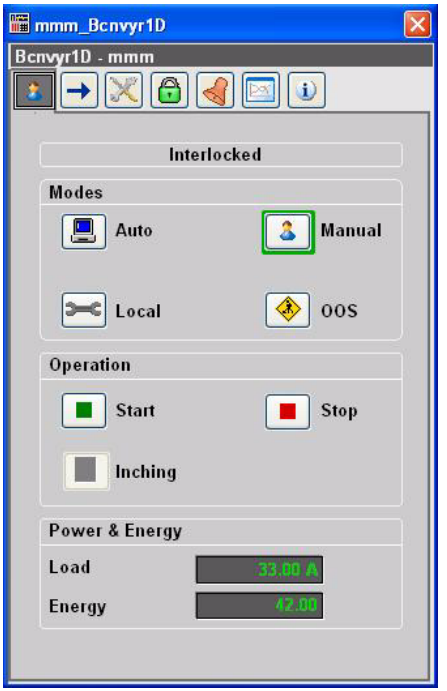
Popup

The `BCnvyr1d` block popup has the following tabs:

- **Operator** tab
- **Extended** tab
- **Configuration** tab
- **Interlock** tab

- **Alarm** tab
- **Trend** tab
- **Information** tab

Operator Tab



This tab has 4 sub-sections:

- **Status**
- **Modes**
- **Operation**
- **Power and Energy**

The sub-section **Status** displays the current status.

The possible displays are listed as follows:

- **Sleeping**
- **Group Active**
- **Ready**
- **Not Ready**
- **Starting**
- **Stopping**
- **Running**
- **Interlocked**

- **Warning**
- **Trip**

The following table describes the sub-section **Modes**:

Item	Description
Auto	Motor runs in Auto mode controlled by the PLC program.
Manual	Motor runs in Manual mode controlled by the HMI.
Local	Motor runs in Local mode controlled by the local control box of the motor.
OOS	Motor runs in Out of Service mode for maintenance purpose of the object (applicable in all modes).

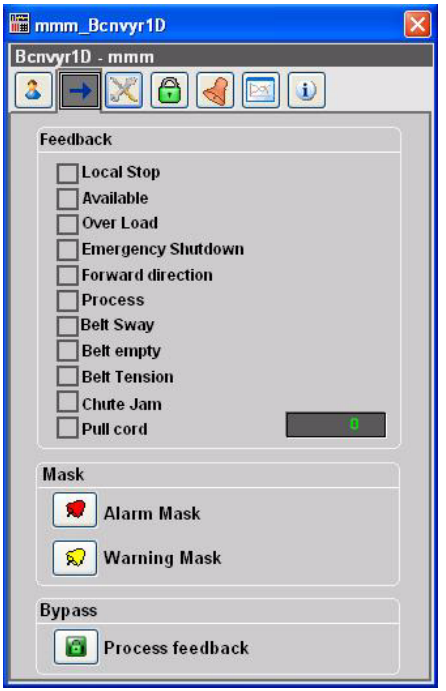
The following table describes the sub-section **Operation**:

Item	Description
Start	Initiates the operation of the device in Manual mode.
Stop	Stops the operation of the device in Manual mode.
Inching	**Inching** command is given to belt conveyor in Local mode.

The following table describes the sub-section **Power and Energy**:

Item	Description
Load	Configurable and displays load of motor in A/kW.
Energy	Configurable and displays energy consumed by motor in kWh.

Extended Tab



This tab has 3 sub-sections:

- **Feedback**
- **Mask**
- **Bypass**

The following table describes the sub-section **Feedback**:

Item	Animation	Description
Local Stop	Steady green	Motor locally not stopped feedback obtained.
	Steady gray	Motor locally stopped feedback obtained.
Available	Steady green	Motor electrically available feedback obtained.
	Steady gray	Motor electrically not available feedback obtained.
Over Load	Steady green	Motor overload feedback obtained.
	Steady gray	Motor overload feedback not obtained.
Emergency shutdown	Steady green	Emergency stop feedback obtained.
	Steady gray	Emergency stop feedback not obtained.
Forward direction	Steady green	Motor is running in forward direction
	Steady gray	Motor is not running in forward direction.

Item	Animation	Description
Process	Steady green	Process feedback is obtained.
	Steady gray	Process feedback is not obtained.
Belt sway	Steady green	Belt sway feedback of belt conveyor is obtained.
	Steady gray	Belt sway feedback of belt conveyor is not obtained.
Belt empty	Steady green	Belt empty feedback of belt conveyor obtained.
	Steady gray	Belt empty feedback of belt conveyor not obtained.
Belt Tension	Steady green	Belt tension feedback of belt conveyor obtained.
	Steady gray	Belt tension feedback of belt conveyor not obtained.
Chute Jam	Steady green	Chute jam feedback of belt conveyor obtained.
	Steady gray	Chute jam feedback of belt conveyor not obtained.
Pull cord	Steady green	Pull cord feedback of belt conveyor obtained.
	Steady gray	Pull cord feedback of belt conveyor not obtained.

The following table describes the Pull cord swich number (**PCN**):

Symbol	Description
	Displays the **PCN** that is pulled out. If Pull cord feedback of belt conveyor is obtained, then 0 is displayed.

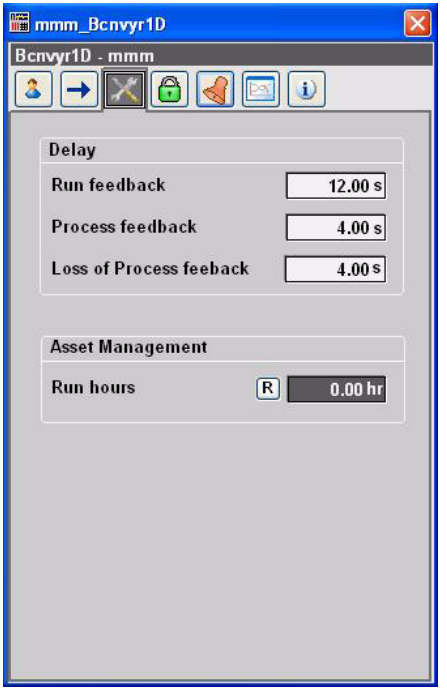
The following table describes the sub-section **Mask**:

Item	Description
Alarm Mask	Masks the indicated current Trip alarms of this block in the alarm list.
Warning Mask	Masks the current alerts of this block in the alarm list.

The following table describes the sub-section **Bypass**:

Item	Description
Bypass	Bypasses the process feedback.

Configuration Tab



This tab has 2 sub-sections:

- **Delay**
- **Asset Management**

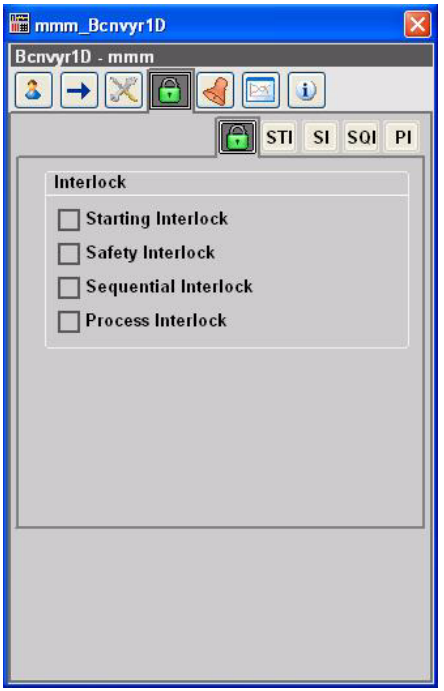
The following table describes the sub-section **Delay**:

Item	Description
Run feedback	Displays run feedback delay of motor configurable from HMI. After starting the motor, if running feedback is not received before delay time, an alarm is generated, and the motor is tripped.
Process feedback	Displays process feedback delay of motor configurable from HMI. After starting the motor, if process feedback is not received before delay time, an alarm is generated, and the motor is tripped.
Loss of Process feedback	Displays loss of process feedback delay of motor configurable from HMI. After receiving the process feedback, if a loss of feedback is occurred while the motor is running and is not received before delay time, an alarm is generated, and the motor is tripped.

The following table describes the sub-section **Asset Management**:

Item	Description
Run hours	Displays the run hours of the motor.
	Resets the run hours value of the motor to 0.

Interlock Tab



This tab has a sub-section called as **Interlock**, which is described in the following table:

Item	Animation	Description
Starting Interlock	Steady green	**Starting Interlock** is active, and the related device is not blocked.
	Steady gray	**Starting Interlock** is not active, and the related device is blocked.

Item	Animation	Description
Safety Interlock	Steady green	**Safety Interlock** is active, and the related device is not blocked.
	Steady gray	**Safety Interlock** is not active, and the related device is blocked.
Sequential Interlock	Steady green	**Sequential Interlock** is active, and the related device is not blocked.
	Steady gray	**Sequential Interlock** is not active, and the related device is blocked.
Process Interlock	Steady green	**Process Interlock** is active, and the related device is not blocked.
	Steady gray	**Process Interlock** is not active, and the related device is blocked.

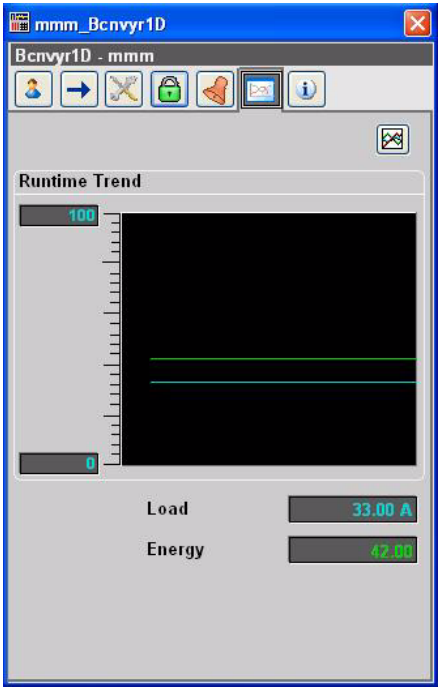
This tab has sub-tabs Interlock, **STI**, **SI**, **SQI**, and **PI**.

For description about the **Interlock** sub-tabs, refer to Commonly Used Tabs *(see page 53)*.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Trend Tab



This tab has 3 sub-sections:

- **Trend Popup**
- **Runtime Trend**
- **Load**
- **Energy**

The sub-section **Runtime Trend** displays the run time trend of the parameters in the control module.

Item	Description
Load	Displays the runtime trend of load.
Energy	Displays the runtime trend of energy.

Item	Description
	Displays the historical trend page.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

BCnvyr1d_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type BCnvyr1d_St for the BCnvyr1d block:

Element	Type	Description
Load	REAL	Load of motor in A/kW
Energy	REAL	Energy consumed by motor in kWh/MWh
RunHr	INT	Run hours of motor
TRunFb	INT	Run feedback delay
TProcFb	INT	Process feedback delay
TLProcFb	INT	Loss process feedback delay
PCN	INT	Pull cord number
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Stw2	WORD	Status word 2 to interact with HMI
Stw3	WORD	Status word 3 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI
Alw2	WORD	Alarm word 2 to interact with HMI
Dummy	WORD	Not used

Bit description for the ctw element:

Bit	Description
0	Auto mode selection from HMI
1	Manual mode selection from HMI
2	Local mode selection from HMI
3	Out of Service mode selection from HMI
4	**Start** command from HMI
5	Not used
6	**Stop** command from HMI
7	Alarm reset from HMI
8	Run hour reset from HMI
9	Bypass process feedback from HMI
10	Inching
11...13	Not used

Bit	Description
14	Alert mask
15	Alarm mask

Bit description for the `Stw1` element:

Bit	Description
0	Auto mode selection from HMI
1	Local mode selection from HMI
2	Group wake
3	Local stop
4	Starting interlock
5	Safety interlock
6	Sequential interlock
7	Process interlock
8	Motor electrically available
9	Motor Overload
10	Emergency stop
11	Run feedback
12	Not used
13	Process feedback
14	Alarm inhibition
15	Not used

Bit description for the `Stw2` element:

Bit	Description
0	Auto mode active status
1	Manual mode active status
2	Local mode active status
3	Out of Service mode active status
4	Output command (**OP**) active
5	Not used
6	Motor ready status
7	Motor running status
8...11	Not used
12	Interlock active status

Bit	Description
13	Group active status
14	Alert active status
15	Trip active status

Bit description for the `Stw3` element:

Bit	Description
0	**Belt Sway** switch
1	**Belt Empty Sensor** switch
2	**Pull Cord** switch
3	**Belt Tension** switch
4	**Chute Jam** switch
5...15	Not used

Bit description for the `Alw1` element:

Bit	Description
0	Motor locally stopped
1	Safety interlock not available
2	Sequential interlock not available
3	Process interlock not available
4	Motor electrically not available
5	Motor overload
6	Motor emergency stop
7	Run feedback not active when motor is running
8	Run feedback active when motor is not running
9...10	Not used
11	Process feedback not active when motor is running
12	Process feedback active when motor is not running
13	Not used
14	Run hours maximum value reached
15	Active trip alarm

Bit description for the `Alw2` element:

Bit	Description
0	**Belt Sway** switch not available
1	**Belt Empty Sensor** switch not available
2	**Pull Cord** switch not available
3	**Belt Tension** switch not available
4	**Chute Jam** switch not available
5...15	Not used

9.5 RouteMgt: Route Management

Overview

This section describes the RouteMgt block.

What's in this Section?

This section contains the following topics:

Topic	Page
RouteMgt - Description	419
RouteMgt - Representation as SCoD in UAG	427
RouteMgt - Representation in the HMI	430
RouteMgt_St Derived Data Type	436

RouteMgt - Description

General

The RouteMgt block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB RouteMgt
Vijeo Citect	Genie RouteMgt
UAG	SCoD RouteMgt

Function Description

Different sets of devices that are belonging to same operational group, performing similar functionality and are mutually exclusive are called as routes (for example, transportation of different grade cement to respective silo involving different sets of actuators).

This block Is used to make selection between routes and to control the start and stop of devices of a selected route.

Block supports upto 5 routes. The total number of active routes need to be configured (RouteCount). During initialization, pre-defined default route (DRoute) is selected. Later, the route can be selected by the operator from HMI popup or by PLC program pin (RouteSel).

Block wake-up command (**Wake**) is given by the GrpCtrl block. The route slection block is ready to start if the interlocks (StrtILck, SftyILck, SeqILck) are ok, there are no active trip alarms and ready feedback of selected route is active. **Start**/**Stop** command is given to the route selection block either by the GrpCtrl block or previous device in sequence or GrpStrtMgt block. If **Start** command is received, route start command (**RStart**) pulse output - RxStart (x: 1...5) is given to devices of selected route either by daisy chain (connection to first motor in the sequence) or by using management blocks (connection to GrpStrtMgt block). If **Stop** command is received, route stop command (**RStop**) output - RxStop (x: 1..5) is given to the devices of selected route either by daisy chain (connection to first motor in the sequence) or by using management blocks (connection to GrpStrtMgt block).

If a route selection is changed when the route is running, **Start** command is given to the new route and after the run feedback is received, **Stop** command is given to the old route. If the new route is inoperable to start (run feedback is not received), then an alert is generated, and old route is restored.

Interlock Types

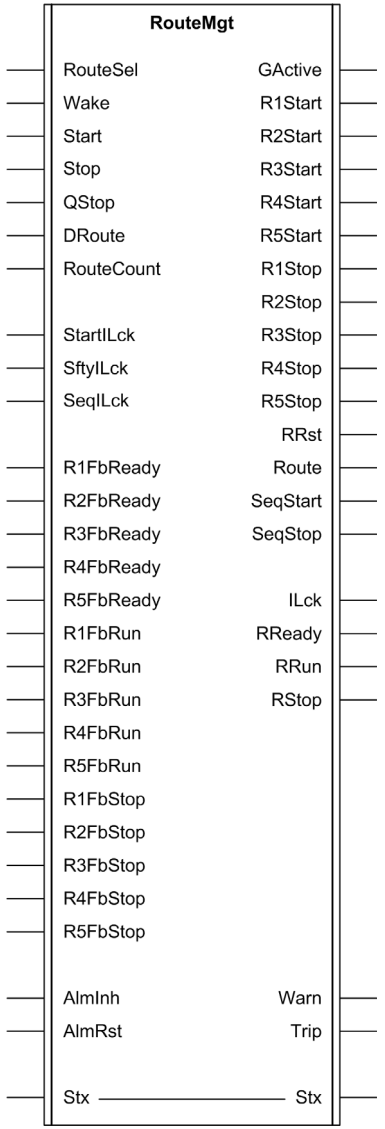
There are 3 types of interlocks. The result of individual interlock condition blocks is connected to the following interlock pins.

Interlock Name	Parameter	Description
Starting Interlock	`StartILck`	These interlocks are valid before start-up only.
Safety Interlock	`SftyILck`	All the safety conditions are taken into consideration.
Sequential Interlock	`SeqILck`	All the sequential conditions are taken into consideration.

Ready, run, stop feedbacks of respective routes are continuously monitored.

Representation in FBD

The following figure shows the FBD representation of the `RouteMgt` block.



Parameter Description

The following table describes the input parameters of the RouteMgt block:

Parameter	Type	Description	Details
RouteSel	INT	Route selected	Selection of route: ● RouteSel (x: 1...5) = 1: Block is forced to Routex (x: 1...5). The route is selected as per the input value. Route selection option in HMI is grayed out. ● RouteSel = 0: Block is set free for operator to selct the route from HMI.
Wake	BOOL	Group wake-up	Group wake-up and request for group availability check: ● Wake = 1: Group is active. ● Wake = 0: Group is in Sleep mode. This input is connected from output GWake of GrpCtrl.
Start	BOOL	Group start	● Start = 1: **Start** command is given to the route management. ● Start = 0: **Start** command is not given to the route management. This input is connected from the Start pin of GrpCtrl or GrpStrtMgt, RouteMgt or previous device in the group.
Stop	BOOL	Group stop	● Stop = 1: **Stop** command is given to the route management. ● Stop = 0: **Stop** command is not given to the route management. This input is connected from the Stop pin of GrpCtrl or GrpStpMgt or RouteMgt or previous device in the group.
Qstop	BOOL	Group quick stop	● QStop = 1: Quick Stop (**QStop**) command to the route management. ● QStop = 0: Quick Stop (**QStop**) command is not given to the route management. This input is connected from the QStop pin of GrpCtrl.
DRoute	INT	Default route	This input defines the default route to be selected when it is initialized.

Parameter	Type	Description	Details
RouteCount	INT	Route count	This input defines the total number of routes present. The block manages routes only between 1 to route count. Route selection for routes more than the route count will be disabled from HMI.
StartILck	BOOL	Starting interlock	• StartILck = 1: All the starting interlock conditions are operational. • StartILck = 0: Not all the starting interlock conditions are operational. Applicable only before the route starts.
SftyILck	BOOL	Safety interlock	• SftyILck = 1: All the safety interlock conditions are operational. • SftyILck = 0: Not all the safety interlock conditions are operational. Trips the route and generates an alarm. **Stop** command is given to all the devices in the route.
SeqILck	BOOL	Sequential Interlock	• SeqILck = 1: All the sequential interlock conditions are operational. • SftyILck = 0: Not all the sequential interlock conditions are not operational. Trips the route and generates an alarm. **Stop** command is given to all the devices in the route.
RxFbReady (x: 1...5)	BOOL	Route x (x: 1...5) feedback ready	• RxFbReady (x: 1...5) = 1: All devices in the route x (x: 1...5) are ready. • RxFbReady (x: 1...5) = 0: Not all devices in the route x (x: 1...5) are ready.
RxFbRun (x: 1...5)	BOOL	Route x (x: 1...5) feedback - all devices	• RxFbRun (x: 1...5) = 1: All devices in the route x (x: 1...5) are running. • RxFbRun (x: 1...5) = 0: Not all devices in the route x (x: 1...5) are running.
RxFbStop (x: 1...5)	BOOL	Route x (x: 1...5) feedback - stop	• RxFbStop (x: 1...5) = 1: All devices in the route x are stopped. • RxFbStop (x: 1...5) = 0: Not all devices in the route x (x: 1...5) are stopped.

⚠ CAUTION
UNINTENDED MACHINE OR PROCESS BEHAVIOR Be sure of the state of security of your machine or process environment. The `AlmInh` parameter to inhibit alarms must be used very carefully. **Failure to follow these instructions can result in injury or equipment damage.**

The following table describes the input parameter of the `RouteMgt` block:

Parameter	Type	Description	Details
`AlmInh`	`BOOL`	Alarm inhibition	• `AlmInh` = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • `AlmInh` = 0: Alarms are not inhibited.

⚠ CAUTION
UNINTENDED MACHINE OR PROCESS BEHAVIOR Be sure of the state of security of your machine or process environment. To reset, alarms must be used very carefully. **Failure to follow these instructions can result in injury or equipment damage.**

The following table describes the input parameter of the `RouteMgt` block:

Parameter	Type	Description	Details
`AlmRst`	`BOOL`	Reset	• `AlmRst` = 1: Resets all the alarms generated by the block. Sends **Reset** command `RRst` = 1 to all the blocks in the route. • `AlmRst` = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from `GRst` pin of `GrpCtrl` block and resets the trip alarm.

The following table describes the output parameters of the RouteMgt block:

Parameter	Type	Description	Details
GActive	BOOL	Group active	• GActive = 1: The route is started in the group and is active. • GActive = 0: The route is not started in the group and is not active.
RxStart (x: 1...5)	BOOL	**Start** command to first motor of route x (x: 1...5)	• RxStart (x: 1...5) = 1: **Start** commmand (pulse output) is given to the first device of route x (x: 1...5) or GrpStrtMgt. • RxStart (x: 1...5) = 0: **Start** commmand is not given.
RxStop (x: 1...5)	BOOL	**Stop** command to first motor of route x (x: 1...5)	• RxStop (x: 1...5) = 1: **Stop** commmand (pulse output) is given to the first device of route x (x: 1...5) or GrpStpMgt. • RxStop (x: 1...5) = 0: **Stop** commmand is not given. This output is connected to the last device in the stop sequence GrpStpMgt.
RRst	BOOL	Route reset	• RRst = 1: Resets alarms and alerts of the device in the route. • RRst = 0: No reset of alarms and alerts. **NOTE:** This output has to be connected to input AlmRst of each block of the route.
Route	INT	Route selected	This gives the value of currently selected route.
SeqStart	BOOL	**Start** command to next motor in sequence	• SeqStart = 1: **Start** commmand (pulse output) is given to the next device connected in the start sequence. • SeqStart = 0: **Start** commmand (pulse output) is not given to the next device connected in the start sequence.

Parameter	Type	Description	Details
SeqStop	BOOL	**Stop** command to next motor in sequence	• SeqStop = 1: **Stop** commmand (pulse output) is given to the next device connected in the stop sequence. • SeqStop = 0: **Stop** commmand (pulse output) is given to the next device connected in the stop sequence.
ILck	BOOL	Interlock status	• Ilck = 1: All the interlocks are operable. • Ilck = 0: Not all the interlocks are operable.
RReady	BOOL	Route is ready to start	• RReady = 1: All the conditions to start the route are operable. • RReady = 0: Not all the conditions to start the route are operable.
RRun	BOOL	Route is running	• RRun = 1: All the devices in the selected route are running. • RRun = 0: Not All the devices in the selected route are running.
RStop	BOOL	Route is stopped	• RStop = 1: All the devices in the route are stopped. • RReady = 0: Not All the devices in the route are stopped.
Warn	BOOL	Alert	• Warn = 1: The route selection block is operational with no active alert present. • Warn = 0: The route selection block is operational with an active alert present.
Trip	BOOL	Trip	• Trip = 1: The route selection block is operational with no active alarm present. • Trip = 0: The route selection block is operational with an active trip alarm present.

The following table describes the input/output parameter of the RouteMgt block:

Parameter	Type	Description
Stx	RouteMgt_St	Status and control of the RouteMgt block. Refer to the Structure to interface with the HMI *(see page 436)*.

`RouteMgt` - Representation as SCoD in UAG

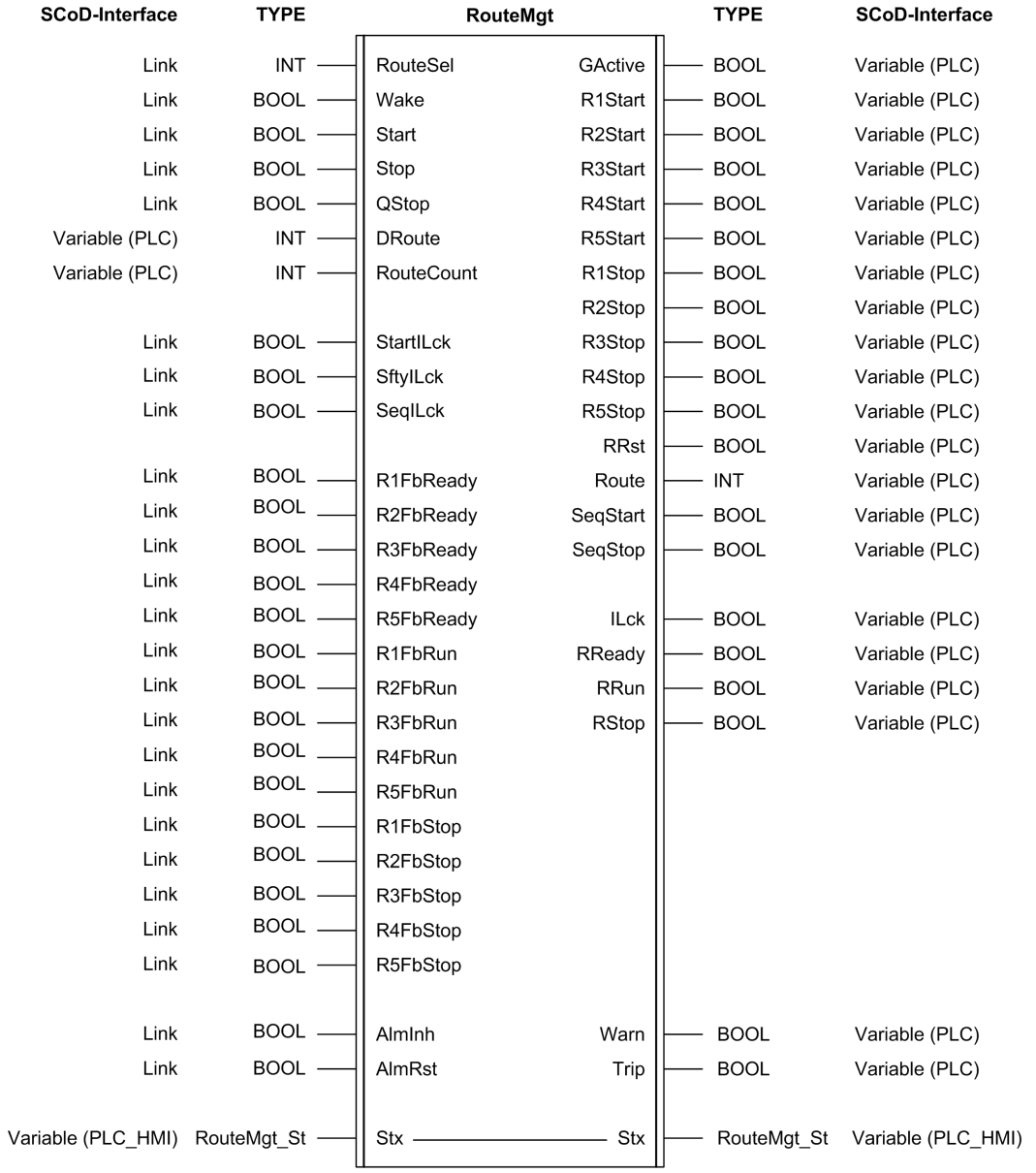
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the `Stx` variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type `HMI` is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the `RouteMgt` block.



SCoD Properties

Name	Name	Initial Value	Property Tab	Position	Description
Default Route	DRoute	0	**Configuration**	8	Default route
Display format (time)	Stx.TStart	0	**Configuration**	1	Display format (time)
Route Count	RouteCount	2- Route 2	**Configuration**	9	Route count
Route Startup Delay	Stx.TStart	0	**Configuration**	2	Route start-up delay
Route Stop Delay	Stx.TStop	0	**Configuration**	3	Route stop delay
Route 1 Name	Route1Desc	Not available	**Route Names**	1	Route 1 name
Route 2 Name	Route2Desc	Not available	**Route Names**	7	Route 2 name
Route 3 Name	Route3Desc	Not available	**Route Names**	2	Route 3 name
Route 4 Name	Route4Desc	Not available	**Route Names**	8	Route 4 name
Route 5 Name	Route5Desc	Not available	**Route Names**	3	Route 5 name

RouteMgt - Representation in the HMI

Route Management (Genie)

The RouteMgt block is represented in the HMI by the following genie.

Genie Elements

Symbol	Meaning
	Opens the RouteMgt popup.
	The instance name of the RouteMgt block is visible by default. You can hide/unhide it by clicking the **Hide/Unhide** button in the template.

Symbol	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Animation Description of the Genie

Symbol	Animation	Meaning
	Line color: Flashing yellow	Alert alarm for the route management block is active.
	Fill color: Steady yellow	Alert alarm for the route management block is acknowledged.
	Line color: Flashing red	Trip alarm for the route management block is active.
	Fill color: Steady red	Trip alarm for the route management block is acknowledged.

Alarm Masking

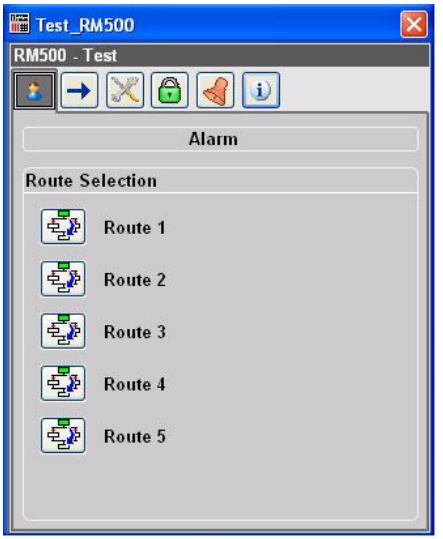
Symbol	Meaning
	The trip or alert alarms in the route management are masked.

Popup

The `RouteMgt` block popup has the following tabs:

- **Operator** tab
- **Extended** tab
- **Configuration** tab
- **Interlock** tab
- **Alarm** tab
- **Information** tab

Operator Tab



This tab has 2 sub-sections:

- **Status**
- **Route Selection**

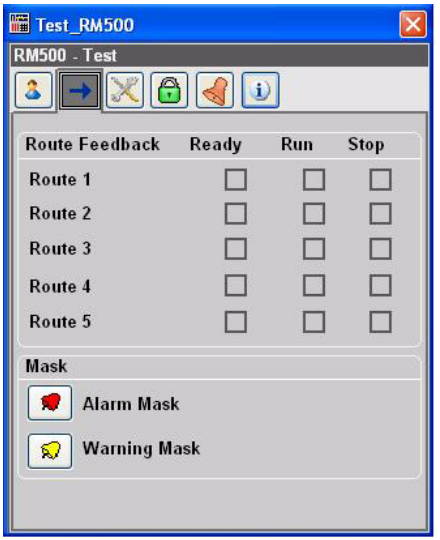
The sub-section **status** displays the current status.

The possible displays are listed as follows:

- **Sleeping**
- **Group Active**
- **Ready**
- **Not Ready**
- **Stop**
- **Running**
- **Interlocked**
- **Warning**
- **Alarm**

This tab displays the number of routes selected in the **Operator** tab.

Extended Tab



This tab has 2 sub-sections:

- **Feedback**
- **Mask**

The following table describes the sub-section **Feedback**:

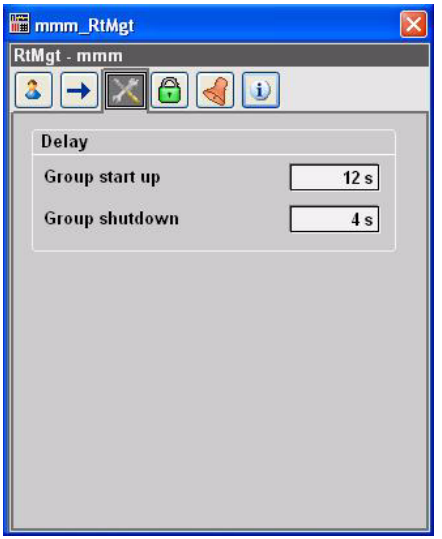
Item	Animation	Description
Ready	Steady green	All devices in the route are ready.
	Light gray	All devices in the route are not ready.

Item	Animation	Description
Run	Steady green	All devices in the route are running.
	Light gray	All devices in the route are not running.
Stop	Steady green	All devices in the route are stopped.
	Light gray	All devices in the route are not stopped.

The following table describes the sub-section **Mask**:

Item	Description
Alarm Mask	Masks the indicated current trip alarms of this block in the alarm list
Warning Mask	Masks the current alerts of this block in the alarm list

Configuration Tab



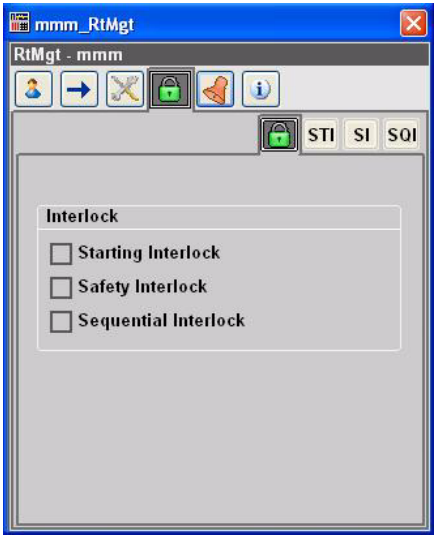
This tab has a sub-section:

- **Delay**

The following table shows the sub-section **Delay**:

Item	Description
Group start up	Displays the group startup time.
Group shutdown	Displays the group stop time.

Interlock Tab



This tab has a sub-section called as **Interlock**, which is described in the following table:

Item	Animation	Description
Starting Interlock	Steady green	**Starting Interlock** is active, and the related device is not blocked.
	Steady gray	**Starting Interlock** is not active, and the related device is blocked.
Safety Interlock	Steady green	**Safety Interlock** is active, and the related device is not blocked.
	Steady gray	**Safety Interlock** is not active, and the related device is blocked.
Sequential Interlock	Steady green	**Sequential Interlock** is active, and the related device is not blocked.
	Steady gray	**Sequential Interlock** is not active, and the related device is blocked.

This tab has sub-tabs Interlock, **STI**, **SI**, and **SQI**.

For description about **Interlock** sub-tabs, refer to Commonly Used Tabs *(see page 53)*.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

RouteMgt_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type RouteMgt_St for the RouteMgt block:

Element	Type	Description
RouteCount	INT	Route count
Route	INT	Route selected
TStart	INT	Route start-up delay
TStop	INT	Route stop delay
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Stw2	WORD	Status word 2 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI

Bit description for the Ctw element:

Bit	Description
0	Route 1 selected
1	Route 2 selected
2	Route 3 select
3	Route 4 selected
4	Route 5 selected
5...6	Not used
7	Alarm reset
8...13	Not used
14	Alarm mask
15	Alert mask

Bit description for the Stw1 element:

Bit	Description
0	Wake
1	Route 1 feedback ready
2	Route 2 feedback ready
3	Route 3 feedback ready
4	Route 4 feedback ready

Bit	Description
5	Route 5 feedback ready
6	Route 1 feedback run
7	Route 2 Feedback run
8	Route 3 feedback run
9	Route 4 feedback run
10	Route 5 feedback run
11	Route 1 feedback stop devices
12	Route 2 feedback stop devices
13	Route 3 Feedback stop devices
14	Alarm inhibition
15	Not used

Bit description for the `Stw2` element:

Bit	Description
0	Route 4 feedback stop devices
1	Route 5 feedback stop devices
2	Start interlock
3	Safety interlock
4	Sequential interlock
5	Not used
6	Group active
7	Interlock
8	Route ready
9	Route run
10	Route forced
11	Route stop
12...13	Not used
14	Alert
15	Trip

Bit description for the `Alw1` element:

Bit	Description
0	Safety Interlock not available
1	Sequential Interlock not available

Bit	Description
2	Start time exceeded
3	Stop time exceeded
4	Route change not available
5	Route selected pin error
6...14	Not used
15	Route trip

9.6 BagFltr: Bag Filter Sequence Block

Overview

This section describes the BagFltr block.

What's in this Section?

This section contains the following topics:

Topic	Page
BagFltr - Description	440
BagFltr - Representation as SCoD in UAG	448
BagFltr - Representation in the HMI	451
BagFltr_St Derived Data Type	458

BagFltr - Description

General

The BagFltr block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB BagFltr
Vijeo Citect	Genie BagFltr
UAG	SCoD BagFltr

Function Description

This is a custom built sequencer block for supervising and controlling the bag filter purging operation. This block supports 18 bags (18 outputs to solenoid valves). This block supports only Auto mode of operation.

Auto Mode Operation

A wake-up command (**Wake**) is given by the GrpCtrl block to wake-up the block from Sleep mode. A group start-up pulse command (**Start**) is given either from the GrpCtrl or GrpStrtMgt or RouteMgt or previous device in the group. After the **Start** command, the block is group active (GActive) and the cycle starts (CycActive). When the cycle is active, a pulse of length (PulWdthTime) is given to bags (OP1 ... OPx, where x = SVCount). A time gap (PulGapTime) is maintained between the purging. A time interval cycle gap time (CycGapTime) is maintained between the purging cycles. If the block is group active, then a pulse command (**SeqStart**) is given to the next motor to start.

The group stop command (**Stop**) or group quick stop command (**QStop**) sets the output command (**OP**) as low. Group stop command (**Stop**) is received by the block from the GrpCtrl or GrpStptMgt or RouteMgt or previous device in the group. Group quick stop command (**QStop**) is received by the block from the GrpCtrl block. If the block is group shut down, then an output command (**SeqStop**) is given to the next motor to stop.

Interlock Types

There are 4 types of interlocks. The result of individual interlock condition blocks is connected to these interlock pins:

Interlock Name	Parameter	Description
Starting Interlock	StartILck	These interlocks are valid before motor start-up only.
Safety Interlock	SftyILck	All the safety conditions are taken into consideration. This input is active in all the modes.

Interlock Name	Parameter	Description
Sequential Interlock	SeqILck	All the sequential conditions are taken into consideration. This input is active only in Auto mode.
Process Interlock	ProcILck	All the process conditions are taken into consideration. This input is active in Auto and Manual mode.

The status of the interlock is displayed in the popup. Detailed description of the interlocks is displayed in the **Interlock** tab of the popup.

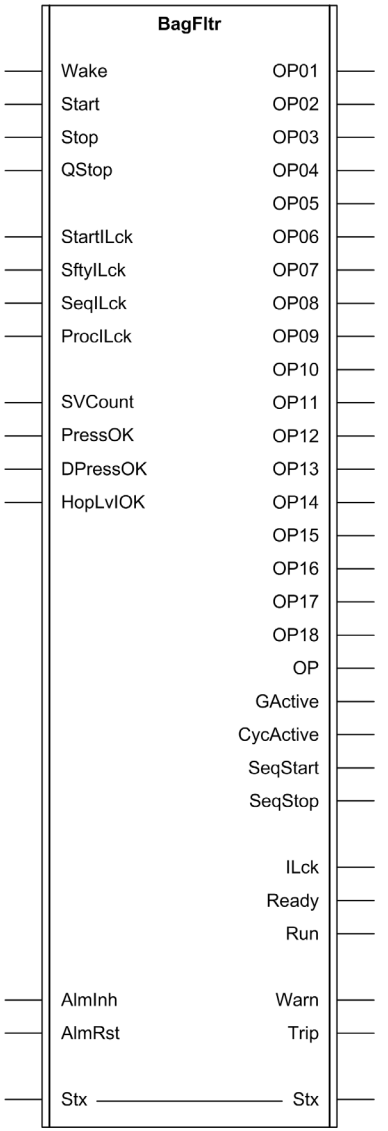
Differential Pressure OK status is connected as input (DPressOK) to the block. If the input is inoperable, then the purging time interval and cycle time interval are halved and an alert is generated.

Hopper Level OK status is connected as input (HopLvlOK) to the block. If the input is inoperable, then an alert is generated.

Pressure OK status is connected as input (PressOK) to the block. If the input is inoperable, then the block is tripped and an alarm is generated. The inoperable conditions (HopLvlOK, PressOK, DPressOK) can be bypassed from HMI popup. The alert and trip is indicated in the SCADA. The details of the active alarms are shown in the alarm tab of the popup. Based on user configured conditions the alarms for this block can be inhibited by connecting the logic to the input pin (AlmInh). The trip alarms can be reset by the input pin (AlmRst) that is connected from GrpCtrl block or by the operator from popup.

Representation in FBD

The following figure shows the FBD representation of the `BagFltr` block.



Parameter Description

The following table describes the input parameters of the `BagFltr` block:

Parameter	Type	Description	Details
`Wake`	`BOOL`	Wake-up	Group wake-up and request for availability check: • `Wake` = 1: Group is active. • `Wake` = 0: Group is in Sleep mode. This input is connected from output `GWake` of `GrpCtrl` block.
`Start`	`BOOL`	Start	• `Start` = 1: **Start** command is given to the bag filter sequencer. • `Start` = 0: **Start** command is not given to the bag filter sequencer. This input is connected from the output pin of `GrpCtrl` or `GrpStrtMgt` or previous device in the group.
`Stop`	`BOOL`	Stop	• `Stop` = 1: **Stop** command is given to the bag filter sequencer. • `Stop` = 0: **Stop** command is not given to the bag filter sequencer. This input is connected from the output pin of `GrpCtrl` or `GrpStpMgt` block or previous device in sequence.
`Qstop`	`BOOL`	Quick stop	• `QStop` = 1: Quick stop command (**QStop**) is given to the bag filter sequencer. • `QStop` = 0: Quick stop command (**QStop**) is not given to the bag filter sequencer. This input is connected from the output pin of `GrpCtrl` block.
`StartILck`	`BOOL`	Starting interlock	• `StartILck` = 1: All the starting interlock conditions are active. • `StartILck` = 0: Not all the starting interlock conditions are active. Applicable only before starting of the bag filter sequencer. After bag filter sequencer is started, `StartILck` is not considered.

Parameter	Type	Description	Details
SftyILck	BOOL	Safety interlock	• SftyILck = 1: All the safety interlock conditions are active. • SftyILck = 0: Not all the safety interlock conditions are active. Trips the bag filter and generates an alarm.
SeqILck	BOOL	Sequential interlock	• SeqILck = 1: All the sequential interlock conditions are active. • SftyILck = 0: Not all the sequential interlock conditions are active. Trips the bag filter and generates an alarm.
ProcILck	BOOL	Process interlock	• ProcILck = 1: All the process interlock conditions are active. • ProcILck = 0: Not all the process interlock conditions are active. Trips the bag filter and generates an alarm.
SVCount	INT	Total bags count in the bag filter	This input is to configure the total no of bags present in the bag filter. The purging cycle starts from OP1 and ends with SVCount.
PressOK	BOOL	Pressure ok	• PressOK = 1: The bag filter is not tripped, and alarm is not generated. Pressure is ok. • PressOK = 0: The bag filter is tripped, and an alarm is generated. Pressure is not ok.

Parameter	Type	Description	Details
DPressOK	BOOL	Differential pressure ok	• DPressOK = 1: The purging time interval and cycle time interval are not halved, and alert is not generated. Differential pressure is ok. • DPressOK = 0: The purging time interval and cycle time interval are halved, and an alert is generated. Differential pressure is not ok.
HopLvIOK	BOOL	Hopper level ok	• HopLvIOK = 1: Alert is not generated. Hopper level is ok. • HopLvIOK = 0: An alert is generated. Hopper level is not ok.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

The AlmInh parameter to inhibit alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the BagFltr block:

Parameter	Type	Description	Details
AlmInh	BOOL	Alarm inhibition	• AlmInh = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • AlmInh = 0: Alarms are not inhibited.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

To reset, alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the BagFltr block:

Parameter	Type	Description	Details
AlmRst	BOOL	Reset	• AlmRst = 1: Resets all the alarms generated by the block. • AlmRst = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from GRst pin of GrpCtrl block and resets the trip alarm.

The following table describes the output parameters of the BagFltr block:

Parameter	Type	Description	Details
OPx (x: 01...18)	BOOL	Output command (**OP**) to purging valve x (x: 01...08)	• OPx (x: 01...18) = 1: Purging valve is given command to open. • OPx (x: 01...18) = 0: Purging valve is given command to close.
OP	INT	Output	This gives the output of current bag or solenoid valve that is getting purged.
GActive	BOOL	Group active	• GActive = 1: The bag filter is started in the group and is active in the group. • GActive = 0: The bag filter is not started in the group and is not active in the group.
CycActive	BOOL	Cycle is active	• CycActive = 1: Cycle is running. • CycActive = 0: Cycle is not running.
SeqStart	BOOL	**Start** command to next motor in sequence	• SeqStart = 1: **Start** commmand (pulse output: SeqStart = 0...1) is given to the next device connected in start sequence. • SeqStart = 0: **Start** commmand (pulse output: SeqStart = 0...1) is not given to the next device connected in start sequence.

Parameter	Type	Description	Details
SeqStop	BOOL	**Stop** command to next motor in sequence	• SeqStop = 1: **Stop** commmand (SeqStop = 1) is given to the next device connected in stop sequence. • SeqStop = 0: **Stop** commmand (SeqStop = 1) is not given to the next device connected in stop sequence.
ILck	BOOL	Interlock status	• ILck = 1: All the interlocks are active. • ILck = 0: Not all the interlocks are active. There is an active interlock present.
Ready	BOOL	Bag filter is ready to start	• Ready = 1: All the conditions to start the bag filter are active. • Ready = 0: Not all the conditions to start the bag filter are active. Bag filter will not be given **Start** command.
Run	BOOL	Bag filter is running	• Run = 1: Bag filter purging operation is on. • Run = 0: Bag filter purging operation is not on. Applicable in all the modes of operation.
Warn	BOOL	Alert	• Warn = 1: The bag filter is operational with no active alert present. • Warn = 0: The bag filter is operational with an active alert present.
Trip	BOOL	Trip	• Trip = 1: The bag filter is operational with no active trip alarm present. • Trip = 0: The bag filter is tripped with an active trip alarm present.

The following table describes the input/output parameter of the BagFltr block:

Parameter	Type	Description
Stx	BagFltr_St	Status and control of the BagFltr block. Refer to the Structure to interface with the HMI *(see page 458)*.

`BagFltr` - Representation as SCoD in UAG

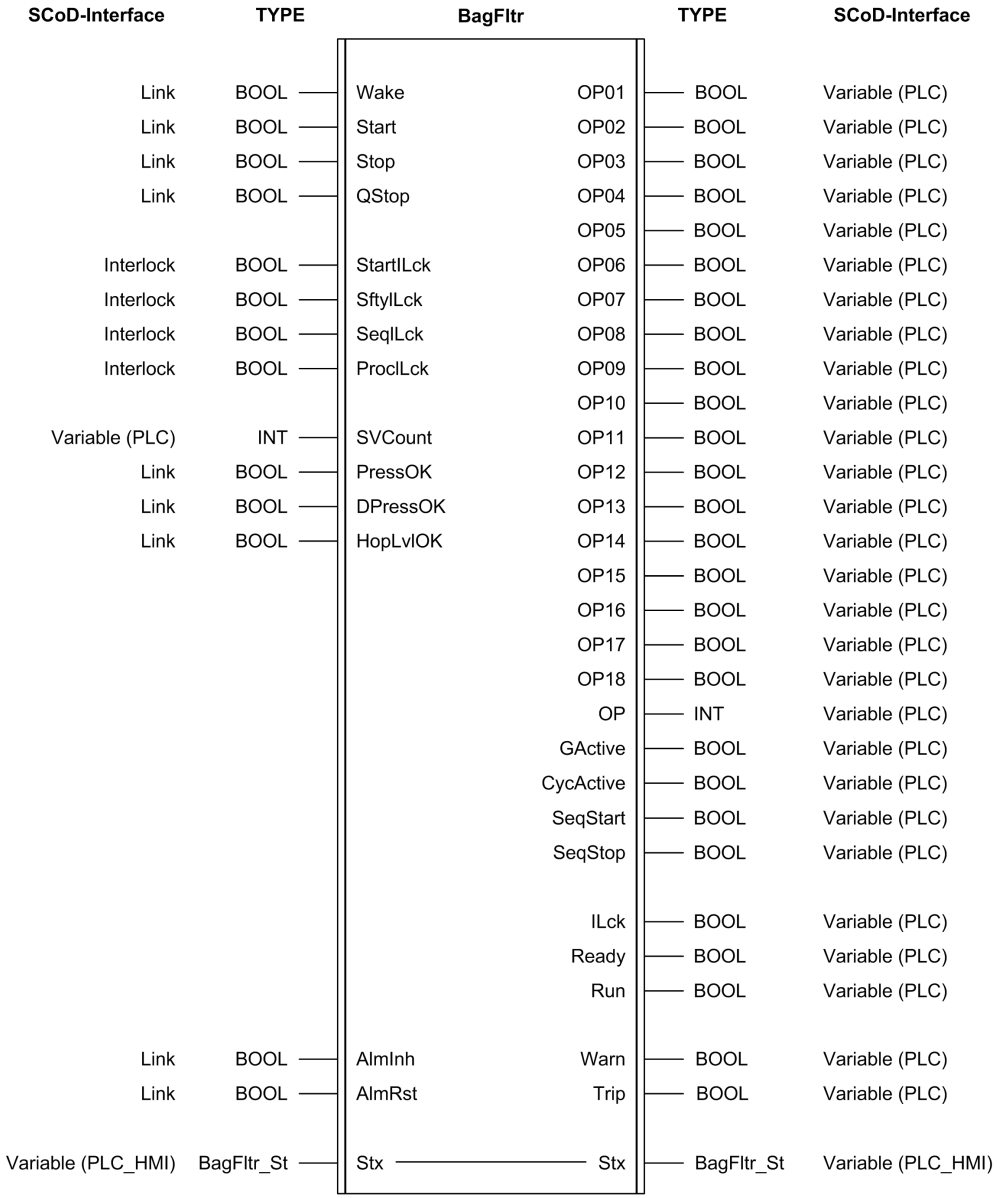
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the `Stx` variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type `HMI` is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the `BagFltr` block.



SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Cycle interval (sec)	`Stx.TCylGap`	0	**Configuration**	3	Cycle interval (sec)
Display format (time)	`Stx.TPulWidth`	0	**Configuration**	1	Display format (time)
Purging Interval (sec)	`Stx.TPulGap`	0	**Configuration**	2	Purging interval (sec)
Purging Time (sec)	`Stx.TPulWidth`	0	**Configuration**	8	Puging time (sec)
Solenoid Valve count	`SVCount`	18	**Configuration**	7	Solenoid valve count
Area Information	`AreaInfo`	Not available	**Device Data**	1	Area information
Asset ID	`AssetID`	Not available	**Device Data**	7	Asset ID

BagFltr - Representation in the HMI

Bag Filter (Genie)

The BagFltr block is represented in the HMI by the following genie.

Genie Elements

Symbol	Meaning
	Opens the BagFltr popup.
	The instance name of the BagFltr block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

Item	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Animation Description of the Genie

Symbol	Animation	Meaning
	Line color: Flashing yellow	Alert alarm for the bag filter block is active.
	Fill color: Steady yellow	Alert alarm for the bag filter block is acknowledged.
	Line color: Flashing red	Trip alarm for the bag filter block is active.
	Fill color: Steady red	Trip alarm for the bag filter block is acknowledged.

Alarm Masking

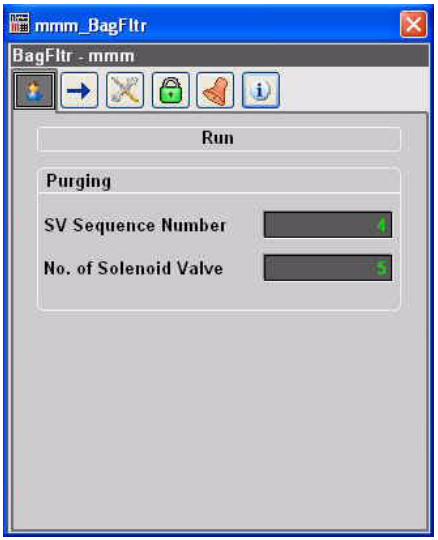
Symbol	Meaning
	The trip or alert alarms in the bag filter are masked.

Popup

The `BagFltr` block popup has the following tabs:

- **Operator** tab
- **Extended** tab
- **Configuration** tab
- **Interlock** tab
- **Alarm** tab
- **Information** tab

Operator Tab



This tab has 2 sub-sections:

- **Status**
- **Purging**

The sub-section **Status** displays the current status.

The possible displays are listed as follows:

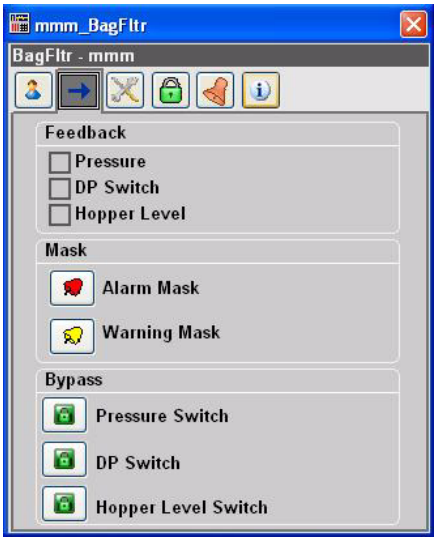
- **Sleeping**
- **Group Active**

- **Ready**
- **Not Ready**
- **Cycle Active**
- **Running**
- **Interlocked**
- **Warning**
- **Alarm**

The following table describes the sub-section **Purging**:

Item	Meaning
SV Sequence Number	Displays the solenoid valve that is purged currently.
No. of Solenoid Valve	Displays the total number of solenoid valves present in the filter.

Extended Tab



This tab has 3 sub-sections:

- **Feedback**
- **Mask**
- **Bypass**

The following table describes the sub-section **Feedback**:

Item	Animation	Description
Pressure	Steady green	**Pressure** feedback is active.
	Steady gray	**Pressure** feedback is not active.
DP Switch	Steady green	**DP Switch** feedback is active.
	Steady gray	**DP Switch** feedback is not active.
Hopper Level	Steady green	**Hopper Level** feedback is active.
	Steady gray	**Hopper Level** feedback is not active.

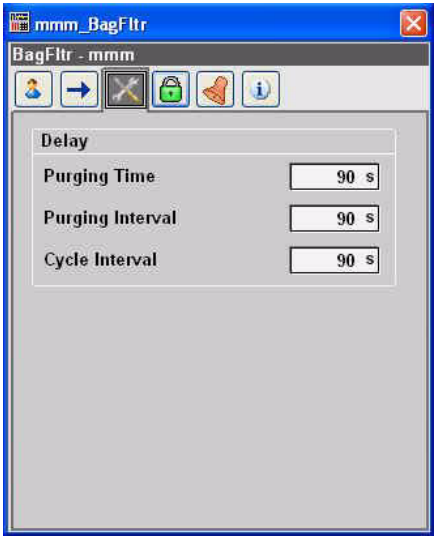
The following table describes the sub-section **Mask**:

Item	Description
Alarm Mask	Masks the indicated current trip alarms of this block in the alarm list.
Warning Mask	Masks the current alerts of this block in the alarm list.

The following table describes the sub-section **Bypass**:

Item	Description
Pressure Switch	**Pressure Switch** inoperable condition monitoring is bypassed.
DP Switch	**DP Switch** inoperable condition monitoring is bypassed.
Hopper Level Switch	**Hopper Level Switch** inoperable condition monitoring is bypassed.

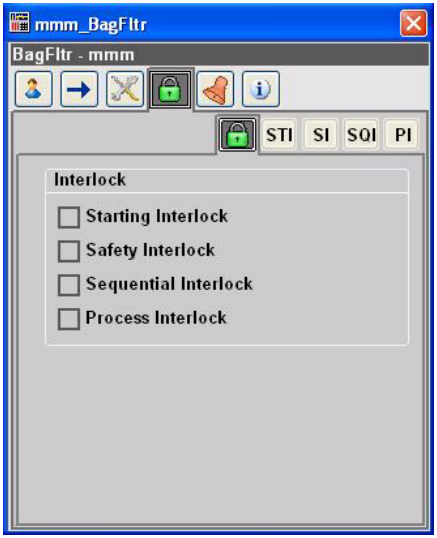
Configuration Tab



This tab has a sub-section called as **Delay**, which is described in the following table:

Item	Description
Purging Time	Displays purging time.
Purging Interval	Displays purging interval.
Cycle Interval	Displays cycle interval.

Interlock Tab



This tab has a sub-section called as **Interlock**, which is described in the following table:

Item	Animation	Description
Starting Interlock	Steady green	**Starting Interlock** is active, and the related device is not blocked.
	Steady gray	**Starting Interlock** is not active, and the related device is blocked.
Safety Interlock	Steady green	**Safety Interlock** is active, and the related device is not blocked.
	Steady gray	**Safety Interlock** is not active, and the related device is blocked.
Sequential Interlock	Steady green	**Sequential Interlock** is active, and the related device is not blocked.
	Steady gray	**Sequential Interlock** is not active, and the related device is blocked.
Process Interlock	Steady green	**Process Interlock** is active, and the related device is not blocked.
	Steady gray	**Process Interlock** is not active, and the related device is blocked.

This tab has sub-tabs Interlock, **STI**, **SI**, **SQI**, and **PI**.

For description about **Interlock** sub-tabs, refer to Commonly Used Tabs *(see page 53)*.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

BagFltr_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type BagFltr_St for the BagFltr block:

Element	Type	Description
SVCount	INT	Solenoid valve count
OP	INT	Output number
TPulWidth	INT	Pulse width time
TPulGap	INT	Pulse gap time
TCylGap	INT	Cycle gap time
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI

Bit description for the Ctw element:

Bit	Description
0	Differential pressure inoperable condition monitoring bypass
1	Pressure inoperable condition monitoring bypass
2	Hopper level inoperable condition monitoring bypass
3...6	Not used
7	Alarm reset
8...13	Not used
14	Alert mask
15	Alarm mask

Bit description for the Stw1 element:

Bit	Description
0	Wake
1	Start interlock
2	Safety interlock
3	Sequential interlock
4	Process interlock
5	Differential pressure ok
6	Pressure ok

Bit	Description
7	Hopper level ok
8	Group active
9	Cycle active
10	Interlock
11	Ready
12	Run
13	Alert
14	Alarm inhibition
15	Trip

Bit description for the `Alw1` element:

Bit	Description
0	Safety interlock not available
1	Sequential interlock not available
2	Process interlock not available
3	Differential pressure inoperable
4	Pressure inoperable
5	Hopper level inoperable
6...14	Not used
15	Bag filter trip

Cement Process

10

Overview

The following chapter describes the Cement process objects.

⚠ WARNING

MISAPPLICATION OF MODULES

The modules in this section do not reflect any specific installation. Before adopting these modules for use in a specific application, the engineer must:

- conduct a safety analysis for the application and equipment installed
- verify that all modules are appropriate for the equipment or function in the installation
- supply appropriate parameters, particularly for limits
- check that all sensors and actuators are compatible with the modules selected
- thoroughly test all functions during verification and commissioning
- provide independent paths for critical control functions (emergency stop, over-limit conditions etc.) according to the safety analysis and applicable codes and regulations.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
10.1	`GasAnalyser`: Gas Analyser	462
10.2	`MultiFuelCtrl`: Multi Fuel Control	487
10.3	`GirthGearLubCtrl`: Girth Gear Lubrication Control	512

10.1 GasAnalyser: Gas Analyser

Overview

This section describes the GasAnalyser block.

What's in this Section?

This section contains the following topics:

Topic	Page
GasAnalyser - Description	463
GasAnalyser - Representation as SCoD in UAG	473
GasAnalyser - Representation in the HMI	476
GasAnalyser_St Derived Data Type	484

GasAnalyser - Description

General

The GasAnalyser block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB GasAnalyser
Vijeo Citect	Genie GasAnalyser
UAG	SCoD GasAnalyser

Function Description

This is a custom built block that supervises and controls the Gas Analyser operation. This block supports 4 inputs (CO, O2, NOx, SOx) and 8 types of analyser.

Operation

The GasAnalyser block is configured for reading the gas parameters (CO, O2, NOx, SOx) based on the type of anaylser configured through the input pin AnalyserType.

This GasAnalyser block reads the following status of the gas analyser operation:

- Measurement operation on (MeasOn)
- Purging operation on (PurgingOn)
- Probe retraction (ProbeRetractOn)
- Scavenging cycle operation on (ScavengingOn)
- Probe inline status (Probe1Inline, Probe2Inline)
- Probe in status (ProbeIn)
- Probe out status (ProbeOut)

The analysed values are read from the analyser through input pins, and are hold when the gas analyser goes to Purging mode. This processed value is given out as output pin and the same is displayed in the popup.

Inoperable Conditions

There are 12 inoperable conditions:

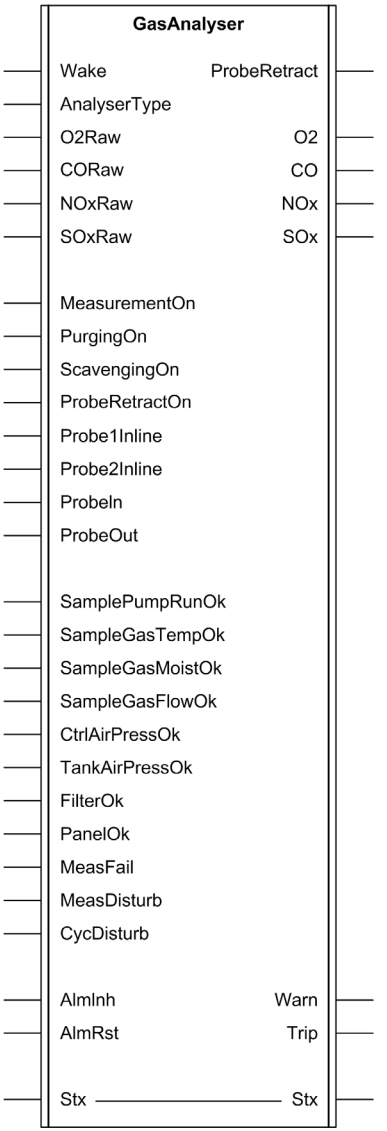
Interlock Name	Parameter	Description
Sample pump run	SamplePumpRunOk	• SamplePumpRunOk = 1: Sample pump run condition is active. • SamplePumpRunOk = 0: Sample pump run condition is not active and an alarm is generated.
Sample gas temperature	SampleGasTempOk	• SampleGasTempOk = 1: Sample gas temperature condition is active. • SampleGasTempOk = 0: Sample gas temperature condition is not active and an alarm is generated.
Sample gas moisture	SampleGasMoistOk	• SampleGasMoistOk = 1: Sample gas moisture condition is active. • SampleGasMoistOk = 0: Sample gas moisture condition is not active and an alarm is generated.
Sample gas flow	SampleGasFlowOk	• SampleGasFlowOk = 1: Sample gas flow condition is active. • SampleGasFlowOk = 0: Sample gas flow condition is not active and an alarm is generated.
Control air pressure	CtrlAirPressOk	• CtrlAirPressOk = 1: Control air pressure condition is active. • CtrlAirPressOk = 0: Control air pressure condition is not active and an alarm is generated.
Tank air pressure	TankAirPressOk	• TankAirPressOk = 1: Tank air pressure condition is active. • TankAirPressOk = 0: Tank air pressure condition is not active and an alarm is generated.

Interlock Name	Parameter	Description
Filter status	`FilterOk`	• `FilterOk` = 1: Filter status is active. • `FilterOk` = 0: Filter status is not active and an alarm is generated.
Panel status	`PanelOk`	• `PanelOk` = 1: Panel status is active. • `PanelOk` = 0: Panel status is not active and an alarm is generated.
Measurement inoperable	`MeasFail`	• `MeasFail` = 1: Measurement is operable. • `MeasFail` = 0: Measurement is inoperable and an alarm is generated.
Measurement disturbance	`MeasDisturb`	• `MeasDisturb` = 1: Measurement disturbance condition is operable. • `MeasDisturb` = 0: Measurement disturbance condition is inoperable and an alert is generated.
Cycle disturbance	`CycDisturb`	• `CycDisturb` = 1: Cycle disturbance condition is operable. • `CycDisturb` = 0: Cycle disturbance condition is inoperable and an alert is generated.

Alarm generated if probe retraction command is given and probe out feedback signal has not come even after a delay time `Stx.TRetract`.

Representation in FBD

The following figure shows the FBD representation of the `GasAnalyser` block.



Parameter Description

The following table describes the input parameters of the GasAnalyser block:

Parameter	Type	Description	Details
Wake	BOOL	Group wake-up	Group wake-up and request for availability check: • Wake = 1: Group is active. • Wake = 0: Group is in Sleep mode. This input is connected from output GWake of GrpCtrl block.
AnalyserType	INT	Analyser Type	AnalyserType = 1: Measures only O2. AnalyserType = 2: Measures only CO. AnalyserType = 3: Measures only NOx. AnalyserType = 4: Measures only SOx. AnalyserType = 5: Measures only O2 and CO. AnalyserType = 6: Measures only NOx and SOx. AnalyserType = 7: Measures only O2, CO and NOx. AnalyserType = 8: Measures all parameters.
O2Raw	REAL	O2 measurement input	O2 value from the analyser.
CORaw	REAL	CO measurement input	CO value from the analyser.
NOxRaw	REAL	NOx measurement input	NOx value from the analyser.
SOxRaw	REAL	SOx measurement input	SOx value from the analyser.
MeasurementOn	BOOL	Measurement on	• MeasurementOn = 1: Analyser is in Measurement mode. • MeasurementOn = 0: Analyser is not in Measurement mode.
PurgingOn	BOOL	Purging on	• PurgingOn = 1: Analyser is in Purging mode. • PurgingOn = 0: Analyser is not in Purging mode.

Parameter	Type	Description	Details
ScavengingOn	BOOL	Scavenging on	• ScavengingOn = 1: Analyser is in Scavenging mode. • ScavengingOn = 0: Analyser is not in Scavenging mode.
ProbeRetractOn	BOOL	Probe retraction on	• ProbeRetractOn = 1: Probe is retracted and analyser is not measuring. • ProbeRetractOn = 0: Analyser is not retracted.
Probe1Inline	BOOL	Probe 1 inline	• Probe1Inline = 1: Probe 1 is inline. • Probe1Inline = 0: Probe 1 is not inline.
Probe2Inline	BOOL	Probe 2 inline	• Probe2Inline = 1: Probe 2 is inline. • Probe2Inline = 0: Probe 2 is not inline.
ProbeIn	BOOL	Probe is in	• ProbeIn = 1: Probe in status is active. • ProbeIn = 0: Probe in status is not active.
ProbeOut	BOOL	Probe is out	• ProbeOut = 1: Probe out status is active. • ProbeOut = 0: Probe out status is not active.
SamplePumpRunOk	BOOL	Sample pump run OK	• SamplePumpRunOk = 1: Sample pump run condition is active. • SamplePumpRunOk = 0: Sample pump run condition is not active and an alarm is generated.
SampleGasTempOk	BOOL	Sample gas temperature OK	• SampleGasTempOk = 1: Sample gas temperature is within normal operating range. • SampleGasTempOk = 0: Sample gas temperature is not within normal operating range and an alarm is generated.

Parameter	Type	Description	Details
SampleGasMoistOk	BOOL	Sample gas moisture OK	• SampleGasMoistOk = 1: Sample gas moisture is within normal operating range. • SampleGasMoistOk = 0: Sample gas moisture is not within normal operating range and an alarm is generated.
SampleGasFlowOk	BOOL	Sample gas flow OK	• SampleGasFlowOk = 1: Sample gas flow is within normal operating range. • SampleGasFlowOk = 0: Sample gas flow is not within normal operating range and an alarm is generated.
CtrlAirPressOk	BOOL	Control air pressure OK	• CtrlAirPressOk = 1: Control air pressure is within normal operating range. • CtrlAirPressOk = 0: Control air pressure is not within normal operating range and an alarm is generated.
TankAirPressOk	BOOL	Tank air pressure OK	• TankAirPressOk = 1: Tank air pressure is within normal operating range. • TankAirPressOk = 0: Tank air pressure is not within normal operating range and an alarm is generated.
FilterOk	BOOL	Filter OK	• FilterOk = 1: Filter status is active. • FilterOk = 0: Filter status is not active and an alarm is generated.
PanelOk	BOOL	Panel OK	• PanelOk = 1: Panel status is active. • PanelOk = 0: Panel status is not active and an alarm is generated.

Parameter	Type	Description	Details
MeasFaillOk	BOOL	Measurement inoperational	• MeasFaillOk = 1: Measurement is operable. • MeasFaillOk = 0: Measurement is inoperable and an alarm is generated.
MeasDisturb	BOOL	Measurement disturbance	• MeasDisturb = 1: Measurement disturbance condition is operable. • MeasDisturb = 0: Measurement disturbance condition is inoperable and an alert is generated.
CycDisturb	BOOL	Cycle disturbance	• CycDisturb = 1: Cycle disturbance condition is operable. • CycDisturb = 0: Cycle disturbance condition is inoperable and an alert is generated.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

The AlmInh parameter to inhibit alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the GasAnalyser block:

Parameter	Type	Description	Details
AlmInh	BOOL	Alarm inhibition	• AlmInh = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • AlmInh = 0: Alarms are not inhibited.

⚠ CAUTION
UNINTENDED MACHINE OR PROCESS BEHAVIOR Be sure of the state of security of your machine or process environment. To reset, all generated alarms must be used very carefully. **Failure to follow these instructions can result in injury or equipment damage.**

The following table describes the input parameter of the `GasAnalyser` block:

Parameter	Type	Description	Details
`AlmRst`	`BOOL`	Reset	• `AlmRst` = 1: Resets all the alarms generated by the block. • `AlmRst` = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from `GRst` pin of `GrpCtrl` block and resets the trip alarm.

The following table describes the output parameters of the `GasAnalyser` block:

Parameter	Type	Description	Details
`ProbeRetract`	`BOOL`	Probe retraction command	• `ProbeRetract` = 1: Probe retraction command is given. • `ProbeRetract` = 0: Probe retraction command is not given.
`O2`	`REAL`	`O2` output	`O2` output
`CO`	`REAL`	`CO`	`CO` output
`NOx`	`REAL`	`NOx` output	`NOx` output
`SOx`	`REAL`	`SOx` output	`SOx` output
`Warn`	`BOOL`	Alert	• `Warn` = 1: The gas analyser is operational with no active alert present. • `Warn` = 0: The gas analyser is operational with an active alert present.
`Trip`	`BOOL`	Trip	• `Trip` = 1: The gas analyser is operational with no active alarm present. • `Trip` = 0: The gas analyser is operational with an active alarm present.

The following table describes the input/output parameter of the `GasAnalyser` block:

Parameter	Type	Description
`Stx`	`GasAnalyser_St`	Status and control of the `GasAnalyser` block. Refer to the Structure to interface with the HMI. *(see page 484)*

GasAnalyser - Representation as SCoD in UAG

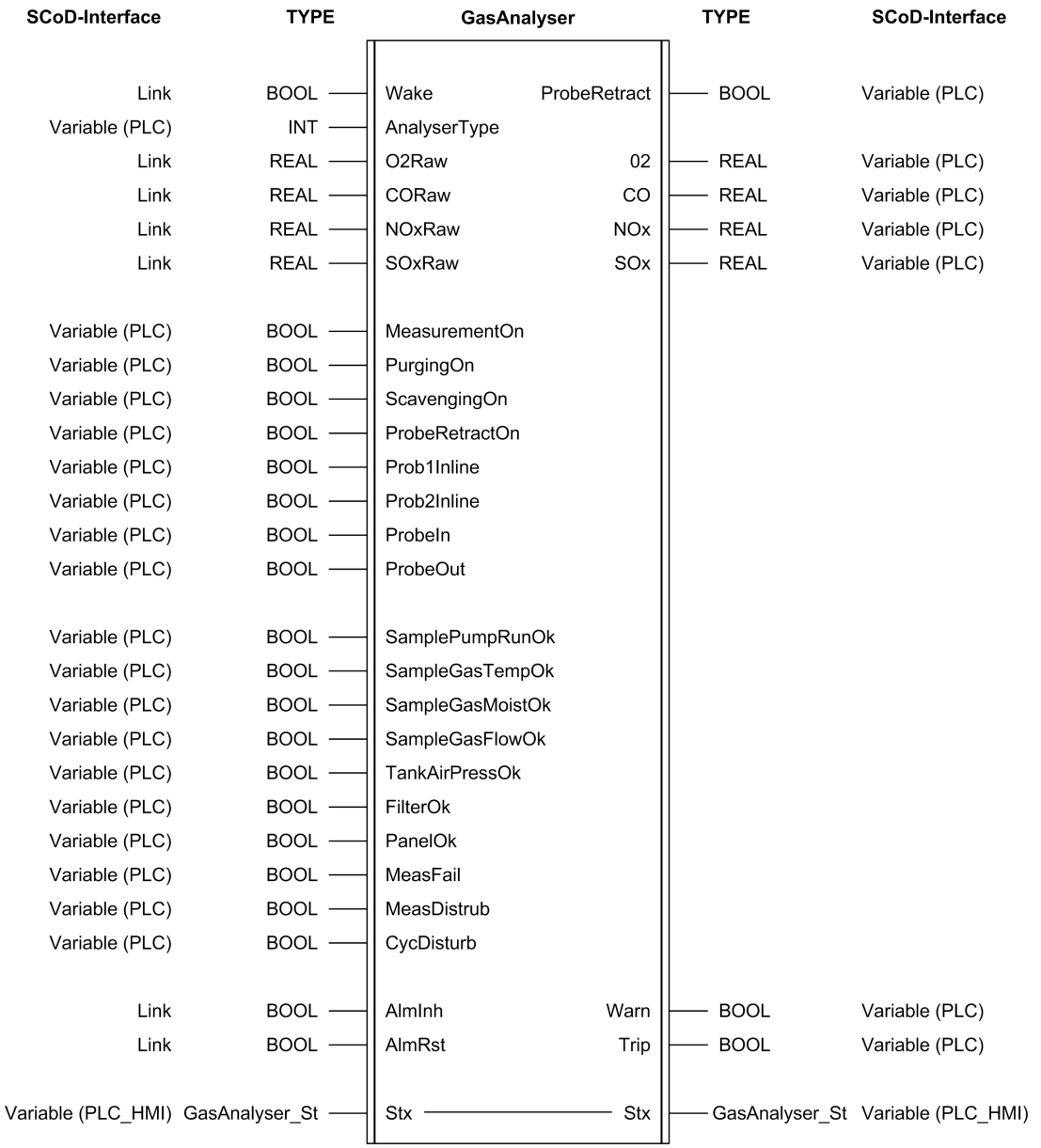
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the `Stx` variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type `HMI` is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the `GasAnalyser` block.



SCoD Properties

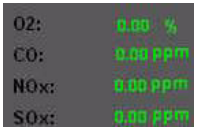
Name	Variable	Initial Value	Property Tab	Position	Description
Analyser type	AnalyserType	1 - O2 measurement enable	**Configuration**	7	Analyser type
Display format (CO)	Stx.CO	Not available	**Configuration**	3	Processed value of CO
Display format (NOx)	Stx.NOx	Not available	**Configuration**	5	Processed value of NOx
Display format (O2)	Stx.O2	Not available	**Configuration**	2	Processed value of O2
Display format (SOx)	Stx.O2	Not available	**Configuration**	4	Processed value of SOx
Display format (time)	Stx.TRetract	0	**Configuration**	1	Time delay for retraction error
Measurement unit (CO)	Stx.CO	Not available	**Configuration**	9	Processed value of CO
Measurement unit (NOx)	Stx.NOx	Not available	**Configuration**	11	Processed value of NOx
Measurement unit (O2)	Stx.O2	Not available	**Configuration**	8	Processed value of O2
Measurement unit (SOx)	Stx.SOx	Not available	**Configuration**	10	Processed value of SOx
Retraction error delay	Stx.TRetract	0	**Configuration**	6	Time delay for retraction error
CO Eng. Min	Stx.CO	Not available	**Range**	2	Processed value of CO
CO Eng. Max	Stx.CO	Not available	**Range**	8	Processed value of CO
NOx Eng. Max	Stx.NOx	Not available	**Range**	10	Processed value of NOx
NOx Eng. Min	Stx.NOx	Not available	**Range**	5	Processed value of NOx
O2 Eng. Max	Stx.O2	Not available	**Range**	7	Processed value of O2
O2 Eng. Min	Stx.O2	Not available	**Range**	1	Processed value of O2
SOx Eng Max	Stx.SOx	Not available	**Range**	9	Processed value of SOx
SOx Eng Min	Stx.SOx	Not available	**Range**	3	Processed value of SOx
Area Information	AreaInfo	Not available	**Device Data**	1	Area Information
Asset ID	AssetID	Not available	**Device Data**	7	Asset ID

`GasAnalyser` - Representation in the HMI

Gas Analyser (Genie)

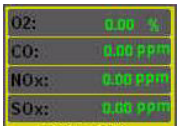
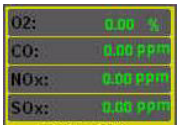
The `GasAnalyser` block is represented in the HMI by the following genie:

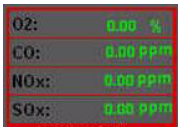
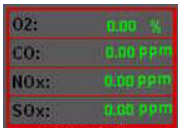
Genie Elements

Symbol	Meaning
	Opens the `GasAnalyser` popup. According to the gas to be analyzed. different genies can be selected.
	The instance name of the `GasAnalyser` block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

Item	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Animation Description of the Genie

Symbol	Animation	Meaning
	Line color: Flashing yellow	Alert for the `GasAnalyser` block is active.
	Line color: Steady yellow	Alert for the `GasAnalyser` block is acknowledged.

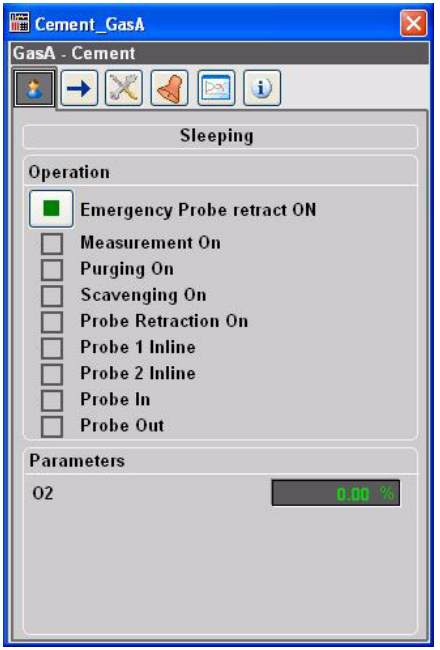
Symbol	Animation	Meaning
	Line color: Flashing red	Alarm for the `GasAnalyser` block is active.
	Line color: Steady red	Alarm for the `GasAnalyser` block is acknowledged.

Popup

The popup of the `GasAnalyser` block has the following tabs:

- **Operator** tab
- **Extended** tab
- **Configuration** tab
- **Alarm** tab
- **Trend** tab
- **Information** tab

Operator Tab



This tab has 3 sub-sections:

- **Status**
- **Operation**
- **Parameters**

The sub-section **Sleeping** displays the current status.

The possible displays are listed as follows:

- **Sleeping**
- **Awake**
- **Warning**
- **Alarm**

The following table describes the sub-section **Operation**:

Item	Animation	Description
Emergency Probe retract ON	Line color: Steady green	Emergency probe retraction command is given to the analyser during emergency situations.
Measurement On	Line color: Steady green	Analyser is in Measurement mode. Probe measurement operation is on.
	Line color: Steady gray	Analyser is not in Measurement mode.

Item	Animation	Description
Purging On	Line color: Steady green	Analyser is in Purging mode.
	Line color: Steady gray	Analyser is not in Purging mode.
Scavenging On	Line color: Steady green	Analyser is in Scavenging mode.
	Line color: Steady gray	Analyser is not in Scavenging mode.
Probe Retraction On	Line color: Steady green	Probe is retracted and analyser is not measuring.
	Line color: Steady gray	Probe is not retracted.
Probe 1 Inline	Line color: Steady green	Probe 1 is inline.
	Line color: Steady gray	Probe 1 is not inline.
Probe 2 Inline	Line color: Steady green	Probe 2 is inline.
	Line color: Steady gray	Probe 2 is not inline.
Probe In	Line color: Steady green	Probe in status is active.
	Line color: Steady gray	Probe in status is not active.
Probe Out	Line color: Steady green	Probe out status is active.
	Line color: Steady gray	Probe out status is not active.

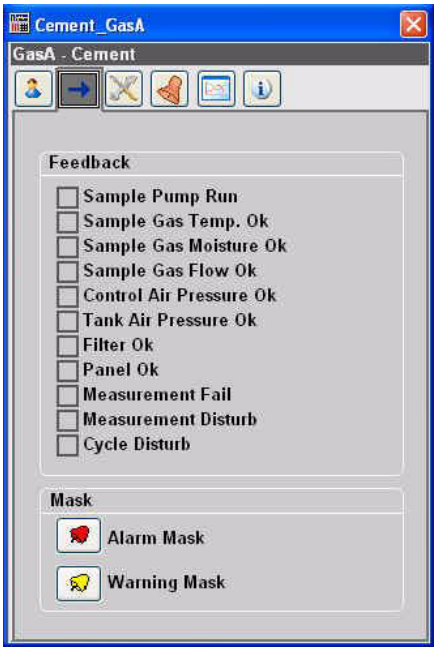
The following table describes the sub-section **Parameters**:

Item	Description
O2	Displays the value of `O2`.

Extended Tab

This tab has 2 sub-sections:

- Feedback
- Mask



The following table describes the sub-section **Feedback**:

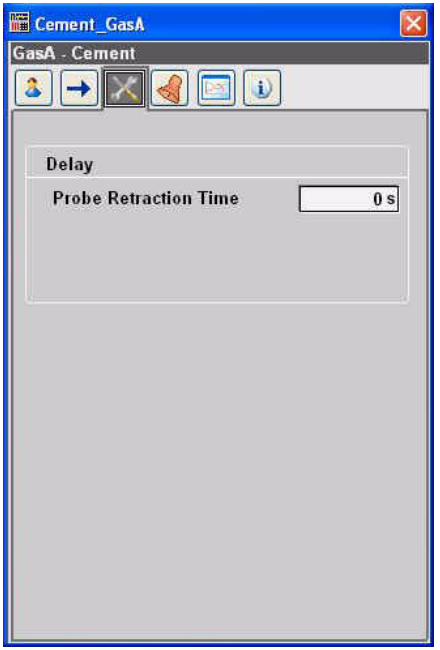
Item	Animation	Description
Sample Pump Run	Steady green	Sample pump run condition is active.
	Steady gray	Sample pump run condition is not active and an alarm is generated.
Sample Gas Temp. Ok	Steady green	Sample gas temperature is within normal operating range.
	Steady gray	Sample gas temperature is not within normal operating range and an alarm is generated.
Sample Gas Moisture Ok	Steady green	Sample gas moisture is within normal operating range.
	Steady gray	Sample gas moisture is not within normal operating range and an alarm is generated.
Sample Gas Flow Ok	Steady green	Sample gas flow is within normal operating range.
	Steady gray	Sample gas flow is not within normal operating range and an alarm is generated.

Item	Animation	Description
Control Air Pressure Ok	Steady green	Control air pressure is within normal operating range.
	Steady gray	Control air pressure is not within normal operating range and an alarm is generated.
Tank Air Pressure Ok	Steady green	Tank air pressure is within normal operating range.
	Steady gray	Tank air pressure is not within normal operating range and an alarm is generated.
Filter Ok	Steady green	Filter status is active.
	Steady gray	Filter status is not active and an alarm is generated.
Panel Ok	Steady green	Panel status is active.
	Steady gray	Panel status is not active and an alarm is generated.
Measurement Fail	Steady green	Measurement is inoperable.
	Steady gray	Measurement is inoperable and an alarm is generated.
Measurement Disturb	Steady green	Measurement disturbance condition is operable.
	Steady gray	Measurement disturbance condition is inoperable and an alert is generated.
Cycle Disturb	Steady green	Cycle disturbance condition is operable.
	Steady gray	Cycle disturbance condition is inoperable and an alert is generated.

The following table describes the sub-section **Mask**:

Item	Description
Alarm Mask	Trip alarms are generated in the alarm list but color animation is not displayed
Warning Mask	Alert alarms are generated in the alarm list but color animation is not displayed.

Configuration Tab



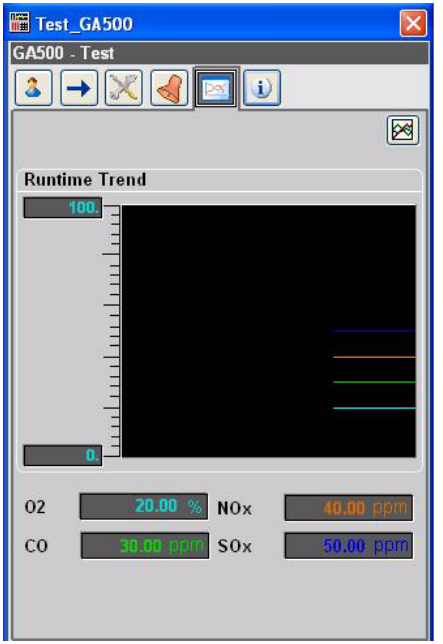
This tab has a sub-section called as **Delay**, which is described in the following table:

Item	Description
Probe Retraction Time	Time delay to generate probe retraction inoperable alarm.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Trend Tab



This tab has 2 sub-sections:

- **Trend Popup**
- **Runtime Trend**

Item	Description
	Displays the associated **Trend** window of the control module.
Runtime Trend	Displays the gas value in **O2**, **CO**, **NOx** and **SOx**.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

GasAnalyser_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type GasAnalyser_St for the GasAnalyser block:

Element	Type	Description
O2	REAL	Oxygen (O2)
CO	REAL	Carbon monoxide (CO)
NOx	REAL	Nitrous oxide (NOx)
SOx	INT	Sulpher oxide (SOx)
Analyser type	INT	Analyser type
TRetract	INT	Time delay for retraction error
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Stw2	WORD	Status word 2 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI

Bit description for the Ctw element:

Bit	Description
0	Probe retract
1...6	Not used
7	Alarm reset
8...13	Not used
14	Alarm mask
15	Alert mask

Bit description for the Stw1 element:

Bit	Description
0	Wake
1	Measurement on
2	Purging on
3	Scavenging on
4	Probe retraction on
5	Probe 1 inline
6	Probe 2 inline

Bit	Description
7	Probe in
8	Probe out
9	Sample pump run ok
10	Sample gas temperature ok
11	Sample gas moisture ok
12	Sample gas flow ok
13	Not used
14	Alarm inhibition
15	Not used

Bit description for the `Stw2` element:

Bit	Description
0	Control air pressure ok
1	Tank air pressure ok
2	Filter ok
3	Panel ok
4	Measurement inoperable
5	Measurement disturbance
6	Cycle disturbance
7...13	Not used
14	Alert
15	Alarm

Bit description for the `Alw1` element:

Bit	Description
0	Probe retract inoperable
1	Sample pump run inoperable
2	Sample gas temperature inoperable
3	Sample gas moist inoperable
4	Sample gas flow inoperable
5	Control air pressure inoperable
6	Tank air pressure inoperable
7	Filter inoperable
8	Panel inoperable

Bit	Description
9	Measurement inoperable
10	Measurement disturbance
11	Cycle disturbance
12...15	Not used

10.2 MultiFuelCtrl: Multi Fuel Control

Overview

This section describes the MultiFuelCtrl block.

What's in this Section?

This section contains the following topics:

Topic	Page
MultiFuelCtrl - Description	488
MultiFuelCtrl - Representation as SCoD in UAG	497
MultiFuelCtrl - Representation in the HMI	503
MFuelCtrl_St Derived Data Type	509

MultiFuelCtrl - Description

General

The MultiFuelCtrl block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB MultiFuelCtrl
Vijeo Citect	Genie MultiFuelCtrl
UAG	SCoD MultiFuelCtrl

Function Description

This MultiFuelCtrl block is used to control the ratio of different fuels of different types.

The main application is managing Kiln Burner Fuel Ratio. This block can support control upto 6 fuels. This block generates 6 set points to individual actuators that indirectly maintains a ratio between 6 different fuels.

This block supports 4 modes of operation:

- **Auto mode 1**
- **Auto mode 2**
- **Manual mode**
- **External mode**

The block goes to Default mode configured (Dmode) during initialization. The block is forced to Auto1 or Auto2 or External mode, if the respective mode selection input pin (Auto1Sel or Auto2Sel or ExtSel) is HIGH.

If the mode selection input pins are LOW, then operator can change the mode from the popup of the block.

Auto Mode 1 Operation

The total set point is fed to the block through input pin (ExtTotalSP) or by the operator from HMI (Stx.ManTotalSP) based on the external set point enabled or disabled.

The individual set point ratios AutoRatiox (x: 1..6) are entered in percentage by the operator from the popup. When the command update ratio (**UpdateRatio**) is given by the operator from HMI and OPxEn (x: 1...6) is enabled, the ratio is updated to the final ratio of the outputs Ratiox (x: 1...6).

Set point Stx.SPxHeat (x: 1...6) is calculated by the formulae Stx.SPxHeat = TotalSP*Ratiox/100. When the ratio or the total set point is changed, the heat set points are adjusted accordingly to maintain the ratio.

The final fuel set point to the respective actuators is calculated by SPx (x: 1…6) = SPxHeat/CalValx (x: 1…6).

The generated set points are restricted to the limits Stx.SPxH and Stx.SPxL (x: 1...6) entered by the operator from HMI. If the total ratio is not equal to 100% for more than pre-defined time (Stx.TRatioError), then an alert is generated.

Auto Mode 2 Operation

This is a kind of expert mode, where the controller defines the ratio of individual fuels based on the priority, high and low limits of the fuel.

The total set point is fed to the block through input pin (ExtTotalSP) or by the operator from HMI (Stx.ManTotalSP) based on the external set point enable/disable.

The operator enters the priority of the fuel and the set point limits from the popup of the MultiFuelCtrl block. The controller calculates these inputs with the inbuilt algorithm and generates the heat set points in GJ/hr. When the ratio or the total set point is changed, the heat set points are adjusted accordingly.

The final fuel set point to the respective actuators is calculated by SPx (x: 1…6) = SPxHeat/CalValx, (x: 1…6).

The generated set points are restricted to the limits Stx.SPxH and Stx.SPxL (x: 1...6) entered by the operator from HMI.

If the total ratio is not equal to 100% for more than pre-defined time (Stx.TRatioError), then an alert is generated.

Manual Mode Operation

When the block is in Manual mode, the manual set point Stx.ManSPx (x: 1...6) is entered by the operator in the popup, and is generated as the output set point SPx (x: 1...6).

Manual set point entry from HMI is restricted to the limits Stx.SPxH and Stx.SPxL (x: 1...6).

External Mode Operation

When the block is in External mode, the external set point is given as an input to the block and is generated as the final fuel set point SPx (x: 1...6) to the respective actuators.

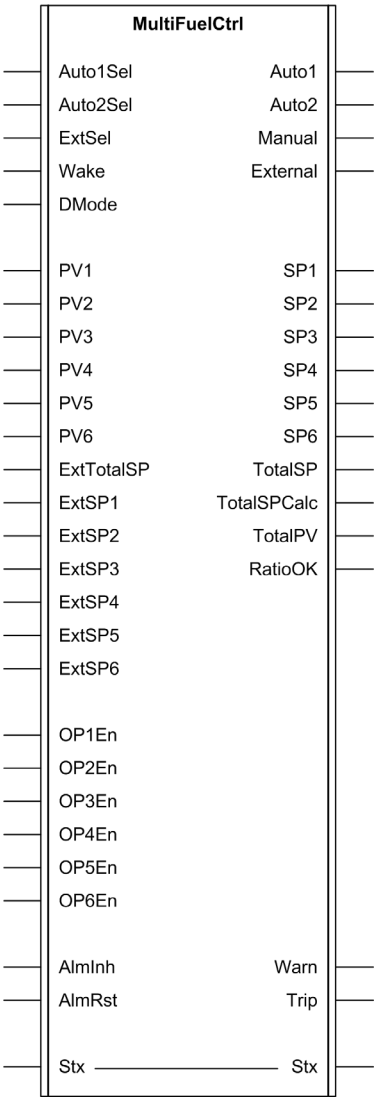
Limits and Inoperable Conditions

The block continuously monitors the process feedback value PVx (x: 1...6). The value of the PVx (x: 1...6) and total process values are displayed in the popup (PV, TotalPV) of the block. If the process value PVx (x: 1...6) deviates from SPx (x: 1…6) beyond limit (STx.DevH) for a longer time (TDev), then an alert is generated.

In the External mode or Auto 1 mode or Auto 2 mode, if the input and output set points are not equal, then an alarm is generated. The alert and alarms are indicated in the SCADA. The details of the active alarms are shown in the **Alarm** tab of the popup.

Representation in FBD

The following figure shows the FBD representation of the `MultiFuelCtrl` block.



Parameter Description

The following table describes the input parameters of the MultiFuelCtrl block:

Parameter	Type	Description	Details
Auto1Selx (x: 1...2)	BOOL	Automatic mode 1 x (x: 1...2) selection	• Auto1Selx (x: 1...2) = 1: Block is forced to auomatic mode 1. Mode selection in popup is grayed out. Set points are generated based on the ratio fixed by operator from popup. • Auto1Selx (x: 1...2) = 0: Block is set free for operator to select the mode from popup.
ExtSel	BOOL	External mode selection	• ExtSel = 1: Set points are generated based on the inputs that comes from other blocks or any other system (For example, Quality control systems). Mode selection option in popup is grayed out. • ExtSel = 0: Block is set free for operator to select the mode from popup.
Wake	BOOL	Group wake-up	Group wake-up and request for availability check: • Wake = 1: Group is active. • Wake = 0: Group is in Sleep mode. This input is connected from output GWake of GrpCtrl block.
DMode	INT	Default mode	• Dmode = 1: Auto1 mode • Dmode = 2: Auto2 mode • Dmode = 3: Manual mode • Dmode = 4: External mode
PVx (x: 1...6)	REAL	Process value feedback x (x: 1...6)	If SPx - PVx> Stx.DevH (x: 1...6) for more than time Stx.TDev, then an alarm is generated.

Parameter	Type	Description	Details
ExtTotalSP	REAL	External total set point	• Applicable in External and Auto modes only if the ExtTotalSPEnable is selected from popup. • In External mode, it is divided into individual set points based on the ratio input pins. In Auto mode, it is divided into individual set points based on the ratio entered by operator from HMI. • Must be connected from other blocks (For example, Systems (Expert control systems) or PID Controller).
ExtRatiox (x: 1...6)	REAL	External ratio for set points x (x: 1...6)	Applicable in External modes only. Set poins are generated based on the external ratio inputs.
OP1Enx (x: 1...6)	BOOL	Output 1...6 x (x: 1...6) enable	• OP1Enx (x: 1...6) = 1: Output x is controlled by the block. • OP1Enx (x:1...6) = 0: Output x is not controlled by the block.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

The AlmInh parameter to inhibit alarms muste be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the MultiFuelCtrl block:

Parameter	Type	Description	Details
AlmInh	BOOL	Alarm inhibition	• AlmInh = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • AlmInh = 0: Alarms are not inhibited.

⚠ CAUTION
UNINTENDED MACHINE OR PROCESS BEHAVIOR Be sure of the state of security of your machine or process environment. To reset, all generated alarms must be used very carefully. **Failure to follow these instructions can result in injury or equipment damage.**

The following table describes the input parameter of the `MultiFuelCtrl` block:

Parameter	Type	Description	Details
`AlmRst`	`BOOL`	Reset	• `AlmRst` = 1: Resets all the alarms generated by the block. • `AlmRst` = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from `GRst` pin of `GrpCtrl` block and resets the trip alarm.

The following table describes the output parameter of the `MultiFuelCtrl` block:

Output	Data Type	Description	Details
`Auto1x` (x: 1...2)	`BOOL`	Automatic mode 1 x (x: 1...2) selected	• `Auto1x` (x: 1...2) = 1: Auto mode 1 is activated. • `Auto1x` (x: 1...2) = 0: Auto mode 1 is not activated.
`Manual`	`BOOL`	Manual mode selected	• `Manual` = 1: Manual mode is activated. • `Manual` = 0: Manual mode is not activated.
`External`	`BOOL`	External mode selected	• `External` = 1: External mode is activated. • `External` = 0: External mode is not activated.

Output	Data Type	Description	Details
SPx (x: 1...6)	REAL	Output set point 1...6	Applicable only if OPx, (x: 1...6) • In External mode, SPx (x: 1...6) = TotalSP * (ExtRatiox (x: 1...6)/100 • In Auto mode, SPx (x:1...6) = TotalSP * (AutoSPx) (x: 1...6)/100 • In Manual mode, SPx = ManSPx (x: 1...6). • Always SPLx < SPx < SPHx (x: 1...6) is maintained
TotalSP	REAL	Total set point output	This is the actual set point given to the ratio controller to generate outputs: ExtTotalSPEnable = 1, TotalSP = ExtTotalSP. ExtTotalSPEnable = 0, TotalSP = Stx.ManTotalSP.
TotaSPcalc	REAL	Total set point	Calculated the total set point .
TotalPV	REAL	Total process value	It is equal to the summation of generated process values PVx (x: 1...6) if the corresponding outputs are enabled.
RatioOK	BOOL	RatioOK	• RatioOK = 1: The sum of ratios enabled outputs is 100%. • RatioOK = 0: The sum of ratios enabled output is not 100%.
Warn	BOOL	Warning	• Warn = 1: The block is operational with no active alert present. • Warn = 0: The block is operational with an active alert present.
Trip	REAL	Trip	• Trip = 1: The block is operational with no active alarm present. • Trip = 0: The block is operational with an active alarm is present.

The following table describes the input/output parameter of the `MultiFuelCtrl` block:

Parameter	Type	Description
`Stx`	`MultiFuelCtrl_St`	Status and control of the `MultiFuelCtrl` block. Refer to the Structure to the Interface with the HMI *(see page 509)*.

`MultiFuelCtrl` - Representation as SCoD in UAG

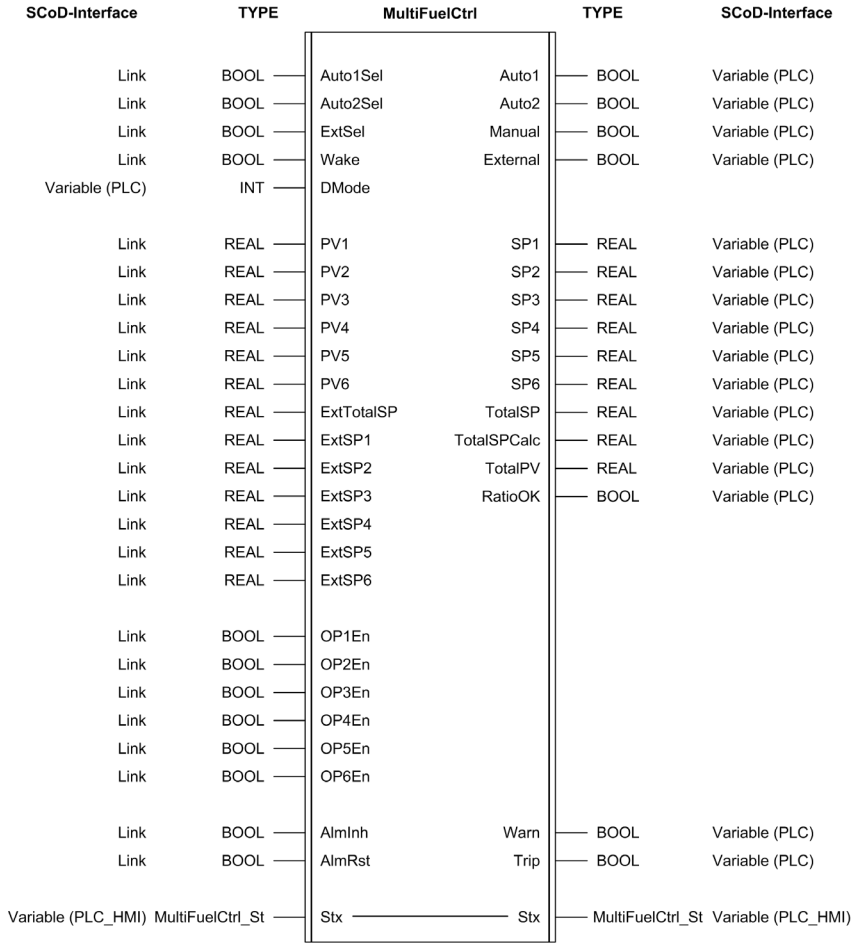
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the `Stx` variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type `HMI` is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the `MultiFuelCtrl` block.



SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Display format (time)	`Stx.TDev`	Not available	**Configuration**	1	Time delay for deviation error
Display format (Total SP)	`Stx.TotalSP`	Not available	**Configuration**	2	Total set point output in GJ/hr

Name	Variable	Initial Value	Property Tab	Position	Description
Display format (SpecificHeat)	Stx.SP1Heat	Not available	**Configuration**	3	Heat set point 1 in GJ/hr
Low limit - Total SP	Stx.TotalSPL	0.0	**Configuration**	4	Total set point low limit in GJ/hr
Deviation high limit	Stx.DevH	0.0	**Configuration**	5	Deviation high limit
Default Mode	DMode	Not available	**Configuration**	7	External mode selection
Measurement Unit (Total SP)	Stx.TotalSP	Not available	**Configuration**	8	Total set point output in GJ/hr
Measurement Unit (SpecificHeat)	Stx.SP1Heat	Not available	**Configuration**	9	Heat set point 1 in GJ/hr
High limit - Total SP	Stx.TotalSPH	100.0	**Configuration**	10	Total set point high limit in GJ/hr
Deviation error delay (min)	Stx.TDev	Not available	**Configuration**	11	Time delay for deviation error
Fuel 1 Name	Fuel1Desc	Not available	**Fuel**	1	Fuel 1 description
Fuel 3 Name	Fuel3Dsec	Not available	**Fuel**	2	Fuel 3 description
Fuel 5 Name	Fuel5Dsec	Not available	**Fuel**	3	Fuel 5 description
Fuel 2 Name	Fuel2Dsec	Not available	**Fuel**	7	Fuel 2 description
Fuel 4 Name	Fuel4Dsec	Not available	**Fuel**	8	Fuel 4 description
Fuel 6 Name	Fuel6Dsec	Not available	**Fuel**	9	Fuel 6 description
Fuel1 Calorific Value	Stx.CalVal1	Not available	**Fuel**	4	Calorific value 1
Fuel3 Calorific Value	Stx.CalVal3	Not available	**Fuel**	5	Calorific value 3
Fuel5 Calorific Value	Stx.CalVal5	Not available	**Fuel**	6	Calorific value 5
Fuel2 Calorific Value	Stx.CalVal2	Not available	**Fuel**	10	Calorific value 2
Fuel4 Calorific Value	Stx.CalVal4	Not available	**Fuel**	11	Calorific value 4
Fuel6 Calorific Value	Stx.CalVal6	Not available	**Fuel**	12	Calorific value 6
Fuel 1 - Low limit SP	Stx.SP1L	Not available	**Limit**	1	Set point 1 low limit
Fuel 2 - Low limit SP	Stx.SP2L	Not available	**Limit**	2	Set point 2 low limit

Name	Variable	Initial Value	Property Tab	Position	Description
Display format (SpecificHeat)	Stx.SP1Heat	Not available	**Configuration**	3	Heat set point 1 in GJ/hr
Low limit - Total SP	Stx.TotalSPL	0.0	**Configuration**	4	Total set point low limit in GJ/hr
Deviation high limit	Stx.DevH	0.0	**Configuration**	5	Deviation high limit
Default Mode	DMode	Not available	**Configuration**	7	External mode selection
Measurement Unit (Total SP)	Stx.TotalSP	Not available	**Configuration**	8	Total set point output in GJ/hr
Measurement Unit (SpecificHeat)	Stx.SP1Heat	Not available	**Configuration**	9	Heat set point 1 in GJ/hr
High limit - Total SP	Stx.TotalSPH	100.0	**Configuration**	10	Total set point high limit in GJ/hr
Deviation error delay (min)	Stx.TDev	Not available	**Configuration**	11	Time delay for deviation error
Fuel 1 Name	Fuel1Desc	Not available	**Fuel**	1	Fuel 1 description
Fuel 3 Name	Fuel3Dsec	Not available	**Fuel**	2	Fuel 3 description
Fuel 5 Name	Fuel5Dsec	Not available	**Fuel**	3	Fuel 5 description
Fuel 2 Name	Fuel2Dsec	Not available	**Fuel**	7	Fuel 2 description
Fuel 4 Name	Fuel4Dsec	Not available	**Fuel**	8	Fuel 4 description
Fuel 6 Name	Fuel6Dsec	Not available	**Fuel**	9	Fuel 6 description
Fuel1 Calorific Value	Stx.CalVal1	Not available	**Fuel**	4	Calorific value 1
Fuel3 Calorific Value	Stx.CalVal3	Not available	**Fuel**	5	Calorific value 3
Fuel5 Calorific Value	Stx.CalVal5	Not available	**Fuel**	6	Calorific value 5
Fuel2 Calorific Value	Stx.CalVal2	Not available	**Fuel**	10	Calorific value 2
Fuel4 Calorific Value	Stx.CalVal4	Not available	**Fuel**	11	Calorific value 4
Fuel6 Calorific Value	Stx.CalVal6	Not available	**Fuel**	12	Calorific value 6
Fuel 1 - Low limit SP	Stx.SP1L	Not available	**Limit**	1	Set point 1 low limit
Fuel 2 - Low limit SP	Stx.SP2L	Not available	**Limit**	2	Set point 2 low limit

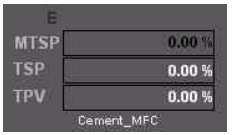
Name	Variable	Initial Value	Property Tab	Position	Description
Fuel 3 - Low limit SP	Stx.SP3L	0.0	**Limit**	3	Set point 3 low limit
Fuel 4 - Low limit SP	Stx.SP4L	0.0	**Limit**	4	Set point 4 low limit
Fuel 5 - Low limit SP	Stx.SP5L	Not available	**Limit**	5	Set point 5 low limit
Fuel 6 - Low limit SP	Stx.SP6L	Not available	**Limit**	6	Set point 6 low limit
Fuel 1 - High limit SP	Stx.SP1H	Not available	**Limit**	7	Set point 1 high limit
Fuel 2 - High limit SP	Stx.SP2H	Not available	**Limit**	8	Set point 2 high limit
Fuel 3 - High limit SP	Stx.SP3H	100.0	**Limit**	9	Set point 3 high limit
Fuel 4 - High limit SP	Stx.SP4H	100.0	**Limit**	10	Set point 4 high limit
Fuel 5 - High limit SP	Stx.SP5H	100.0	**Limit**	11	Set point 5 high limit
Fuel 6 - High limit SP	Stx.SP6H	100.0	**Limit**	12	Set point 6 high limit
Fuel1 Display format (SP)	Stx.SP1	Not available	**Configuration SP**	1	Set point 1
Fuel2 Display format (SP)	Stx.SP2	Not available	**Configuration SP**	2	Set point 2
Fuel3 Display format (SP)	Stx.SP3	Not available	**Configuration SP**	3	Set point 3
Fuel4 Display format (SP)	Stx.SP4	Not available	**Configuration SP**	4	Set point 4
Fuel5 Display format	Stx.SP5	Not available	**Configuration SP**	5	Set point 5
Fuel6 Display format (SP)	Stx.SP6	Not available	**Configuration SP**	6	Set point 6
Fuel1 Measurement Unit (SP)	Stx.SP1	Not available	**Configuration SP**	7	Set point 1
Fuel2 Measurement Unit (SP)	Stx.SP2	Not available	**Configuration SP**	8	Set point 2
Fuel3 Measurement Unit (SP)	Stx.SP3	Not available	**Configuration SP**	9	Set point 3

Name	Variable	Initial Value	Property Tab	Position	Description
Fuel4 Measurement Unit (SP)	`Stx.SP4`	Not available	**Configuration SP**	10	Set point 4
Fuel5 Measurement Unit (SP)	`Stx.SP5`	Not available	**Configuration SP**	11	Set point 5
Fuel6 Measurement Unit (SP)	`Stx.SP6`	Not available	**Configuration SP**	12	Set point 6
Fuel1 CalVal Display format	`Stx.CalVal1`	Not available	**Fuel2**	1	Calorific value 1
Fuel2 CalVal Display format	`Stx.CalVal2`	Not available	**Fuel2**	2	Calorific value 2
Fuel3 CalVal Display format	`Stx.CalVal3`	Not available	**Fuel2**	3	Calorific value 3
Fuel4 CalVal Display format	`Stx.CalVal4`	Not available	**Fuel2**	4	Calorific value 4
Fuel5 CalVal Display format	`Stx.CalVal4`	Not available	**Fuel2**	5	Calorific value 5
Fuel6 CalVal Display format	`Stx.CalVal6`	Not available	**Fuel2**	6	Calorific value 6
Fuel1 CalVal Measurement Unit	`Stx.CalVal1`	Not available	**Fuel2**	7	Calorific value 1
Fuel2 CalVal Measurement Unit	`Stx.CalVal2`	Not available	**Fuel2**	8	Calorific value 2
Fuel3 CalVal Measurement Unit	`Stx.CalVal3`	Not available	**Fuel2**	9	Calorific value 3
Fuel4 CalVal Measurement Unit	`Stx.CalVal4`	Not available	**Fuel2**	10	Calorific value 4
Fuel5 CalVal Measurement Unit	`Stx.Calval5`	Not available	**Fuel2**	11	Calorific value 5
Fuel6 CalVal Measurement Unit	`Stx.CalVal6`	Not available	**Fuel2**	12	Calorific value 6

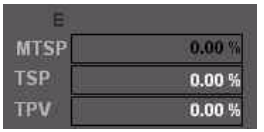
MultiFuelCtrl - Representation in the HMI

Multi Fuel Control (Genie)

The MultiFuelCtrl block is represented in the HMI by the following genie:

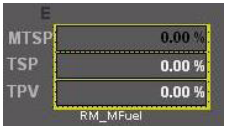
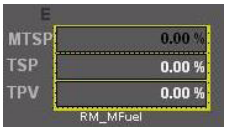
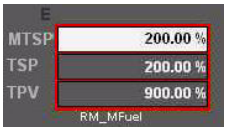
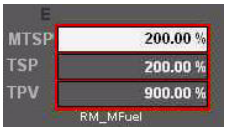


Genie Elements

Symbol	Meaning
	Opens the MultiFuelCtrl popup.
	Displays the manual total set point value of the MultiFuelCtrl block. The value can be entered through genie.
	Displays the total set point value of the MultiFuelCtrl block.
	Displays the total process value of the MultiFuelCtrl block.
	The instance name of the MultiFuelCtrl block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

Item	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Animation Description of the Genie

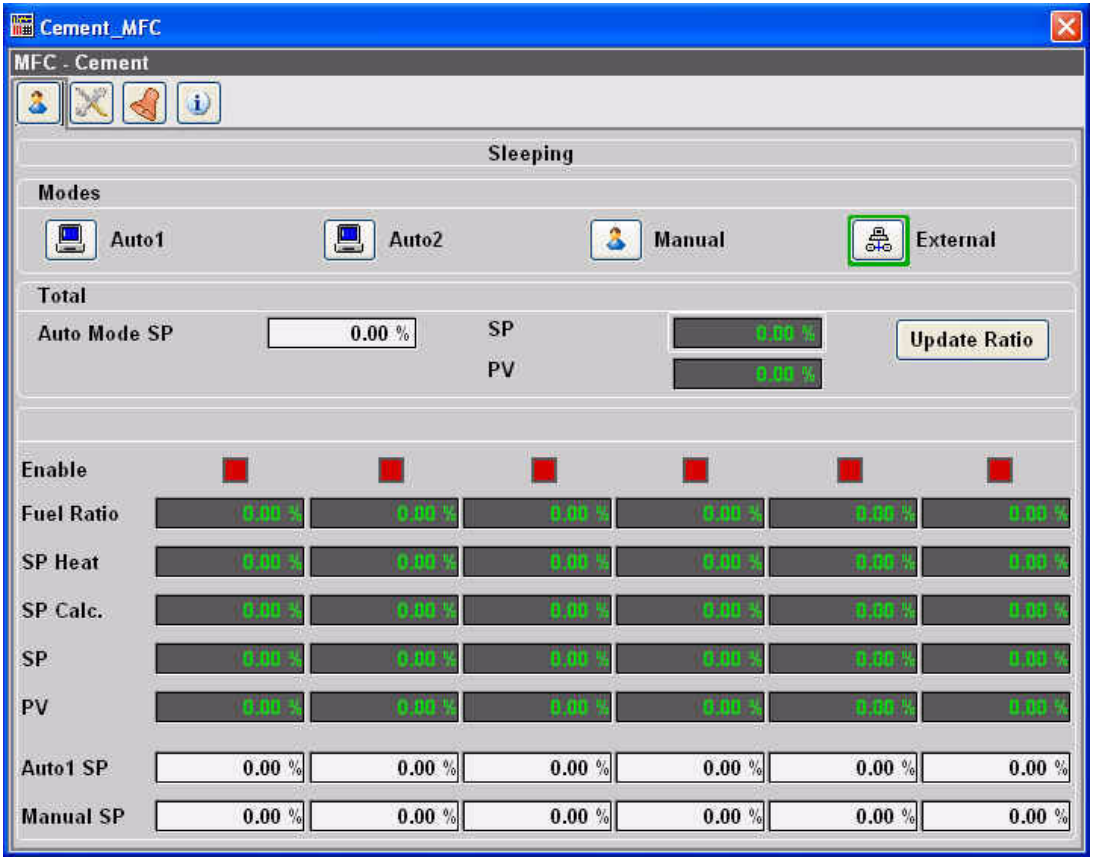
Symbol	Animation	Meaning
	Line color: Flashing yellow	Alert for the `MultiFuelCtrl` block is active and not acknowledged.
	Line color: Steady yellow	Alert for the `MultiFuelCtrl` block is active and acknowledged.
	Line color: Flashing red	Alarm for the `MultiFuelCtrl` block is active and not acknowledged.
	Line color: Steady red	Alarm for the `MultiFuelCtrl` block is active and acknowledged.

Popup

The `MultiFuelCtrl` block popup has the following tabs:

- **Operator** tab
- **Configuration** tab
- **Alarm** tab
- **Information** tab

Operator Tab



This tab has 4 sub-sections:

- **Modes**
- **Total**
- **Outputs**

The sub-section **Status** displays the current status.

The possible displays are listed below:

- **Sleeping**
- **Awake**
- **Ratio Ok**
- **Warning**
- **Alarm**

The following table describes the sub-section **Modes**:

Item	Description
Auto1	The set point output is generated based on ratio (auto set point) entered by the operator from popup.
Auto2	The set point output is generated based on controller's priority based algorithm.
Manual	The set point output is generated based on the value set by the operator from popup.
External	The set point output is generated based on external set points.

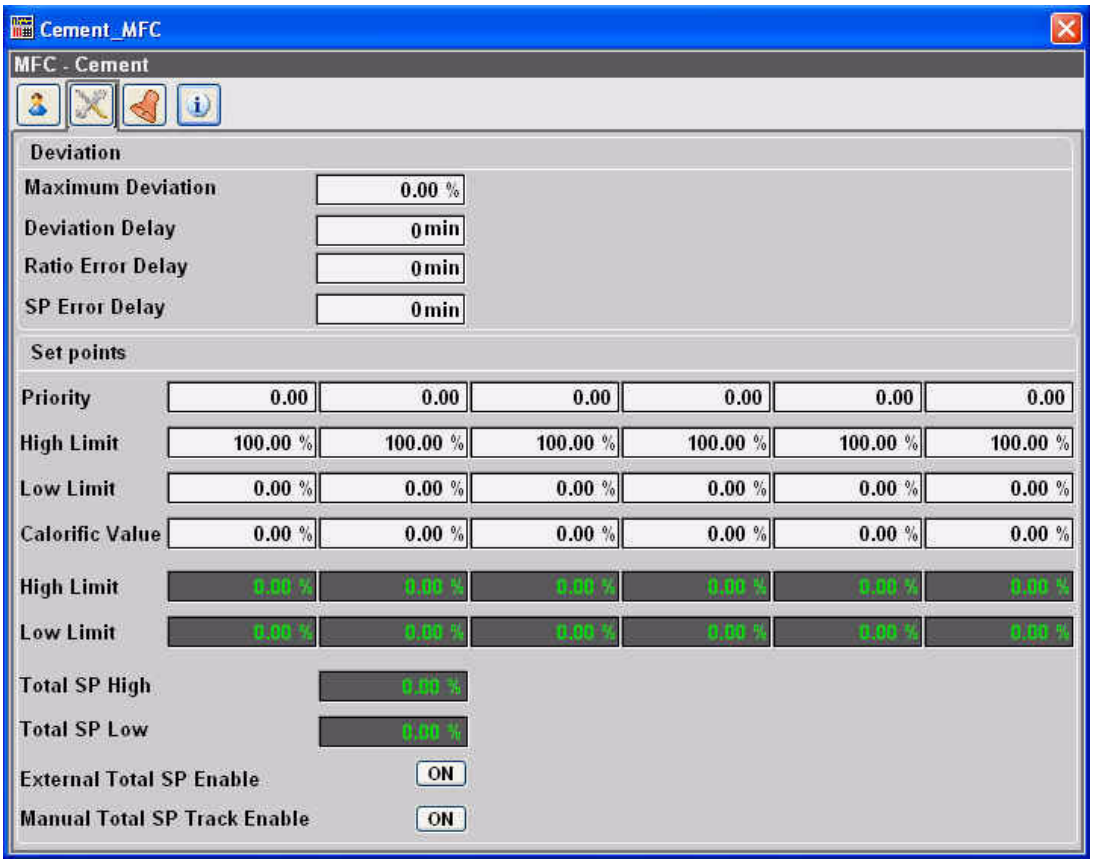
The following table describes the sub-section **Total**:

Item	Description
Auto Mode SP	Displays the total set point entered by the operator.
SP	Displays the total set point value.
PV	Displays the total process value.
Update Ratio	Updates the ratio used for calculation by the operator.

The table below describes the sub-section **Enable** (ratio numbers):

Item	Description
R1...R6	Displays the fuel ratio numbers.
Enable	Displays the ratio output enable status.
Fuel Ratio	Displays the final ratio in %.
SP Heat	Displays the heat set point in GJ/hr.
SP Calc.	Displays the calculated set point
SP	Displays the final set point given as output.
PV	Displays the process value.
Auto1 SP	Displays Auto mode set point (fuel ratio) in %, which is configured from popup.
Manual SP	Displays Manual mode set point, which is configured from popup.

Configuration Tab



This tab has 2 sub-sections:

- **Deviation**
- **Set points**

The table below describes the sub-section **Deviation**:

Item	Description
Maximum Deviation	Displays the maximum deviation.
Deviation Delay	Displays the deviation delay.
Ratio Error Delay	Displays the ratio error delay.
SP Error Delay	Displays the set point error delay.

The table below describes the sub-section **F1…F6**:

Item	Description
Priority	Displays the priority of the fuel.
High Limit	Displays the high limit of the set point.
Low Limit	Displays the low limit of the set point.
Calorific Value	Displays the calorific value of the fuel.
High Limit	Displays the high limit of heat set point.
Low Limit	Displays the low limit of heat set point.
Total SP High	Displays the high limit of total set point.
Total SP Low	Displays the low limit of total set point.
External Total SP Enable	Enable/ disable the external total set point.
Manual Total SP Track Enable	Enable/ disable the manual total set point tracking.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

MFuelCtrl_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type MFuelCtrl_St for the MultiFuelCtrl block:

Element	Type	Description
ManTotalSP	REAL	Manual total set point in GJ/hr
TotalSP	REAL	Total set point output in GJ/hr
TotalSPCalc	REAL	Calculated total set point in GJ/hr
TotalPV	REAL	Total process value in GJ/hr
AutoRatio1...6 (x: 1...6)	REAL	Auto mode ratio set point x (x: 1...6)
Ratio1...6 (x:1...6)	REAL	Final ratio x (x: 1...6)
CalValx (x: 1...6)	REAL	Calorific Value x (x: 1...6)
ManSPx (x: 1...6)	REAL	Manual mode set pioint x (x: 1...6)
ExtSPx (x: 1...6)	REAL	External mode set pioint x (x:1...6)
SPxHeat (x: 1...6)	REAL	Heat set pioint x (x: 1...6) in GJ/Hr
SPxHeatH (x: 1...6)	REAL	Heat set pioint x (x: 1...6) high limit in GJ/Hr
SPxHeatL (x: 1...6)	REAL	Heat set pioint x (x: 1...6) low limit in GJ/Hr
SPxCalc (x: 1...6)	REAL	Calculated set pioint x (x: 1...6)
SPx (x: 1...6)	REAL	Set pioint x (x: 1...6)
SPxH (x: 1...6)	REAL	Set pioint x (x: 1...6) high limit
SPxL (x: 1...6)	REAL	Set pioint x (x: 1...6) low limit
PVx (x: 1...6)	REAL	Process value x (x: 1...6)
TotalSPH	REAL	Total set point high limit in GJ/hr
TotalSPL	REAL	Total set point low limit in GJ/hr
DevH	REAL	Deviation high limit
TDev	INT	Time delay for deviation error
TRatioError	INT	Time delay for ratio error
TSPError	INT	Time delay for set point error
OPxPriority (x: 1...6)	INT	Output x (x: 1...6) priority of usage
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Stw2	WORD	Status word 2 to interact with HMI
Alw	WORD	Alarm word to interact with HMI
Dummy	WORD	Not used

Bit description for the `Ctw` element:

Bit	Description
0	Auto1
1	Auto2
2	Manual
3	External
4	Update ratio
5	External total set point enable
6	Manual total setpoint track enable
7	Alarm reset
8...13	Not used
14	Alarm mask
15	Alert mask

Bit description for the `Stw1` element:

Bit	Description
0	Wake
1	Auto1 selection
2	Auto2 selection
3	Extrenal selection
4	Output 1 enable
5	Output 2 enable
4	Output 3 enable
4	Output 4 enable
4	Output 5 enable
4	Output 6 enable
10...13	Not used
14	Alarm inhibition
15	Not used

Bit description for the `Stw2` element:

Bit	Description
0	Auto1
1	Auto2
2	Manual

Bit	Description
3	External
4	Ratio OK
5...13	Not used
14	Alert
15	Not used

Bit description for the `Alw1` element:

Bit	Description
0	Set point1 limit error
1	Set point2 limit error
2	Set point3 limit error
3	Set point4 limit error
4	Set point5 limit error
5	Set point6 limit error
6	Maximum process value1 deviation
7	Maximum process value2 deviation
8	Maximum process value3 deviation
9	Maximum process value4 deviation
10	Maximum process value5 deviation
11	Maximum process value6 deviation
12	Ratio error
13	More than one mode is selected in input pin
14...15	Not used

10.3 GirthGearLubCtrl: Girth Gear Lubrication Control

Overview

This section describes the GirthGearLubCtrl block.

What's in this Section?

This section contains the following topics:

Topic	Page
GirthGearLubCtrl - Description	513
GirthGearLubCtrl - Representation as SCoD in UAG	519
GirthGearLubCtrl - Representation in the HMI	522
GirthGearLubCtrl_St Derived Data Type	528

GirthGearLubCtrl - Description

General

The GirthGearLubCtrl block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB GirthGearLubCtrl
Vijeo Citect	Genie GirthGearLubCtrl
UAG	SCoD GirthGearLubCtrl

Function Description

This is a custom built block for supervising and controlling the girth gear lubrication operation. This blocks supports 8 spray nozzles and monitors the inoperable conditions.

Auto Mode Operation

A wake-up command (**Wake**) is given by the GrpCtrl block to wake-up the block from Sleep mode. The GirthGearLubCtrl block takes the open and closed feedbacks of valves (Valve1 and Valve2) and grease spray and grease fill feedback of nozzles (nozzle 1A, 2A, 3A, 4A, 1B, 2B, 3B, and 4B). Valve 1 serves nozzles 1A, 1B, 2A, 2B and Valve 2 serves Nozzles 3A, 3B, 4A, and 4B. The block checks for the synchronization of nozzle spray feedback with valve open feedback and nozzle fill feedback with valve close feedback, and generates the status as operable. If all the nozzle status are operable, then it generates the status of the cycle to be OK.

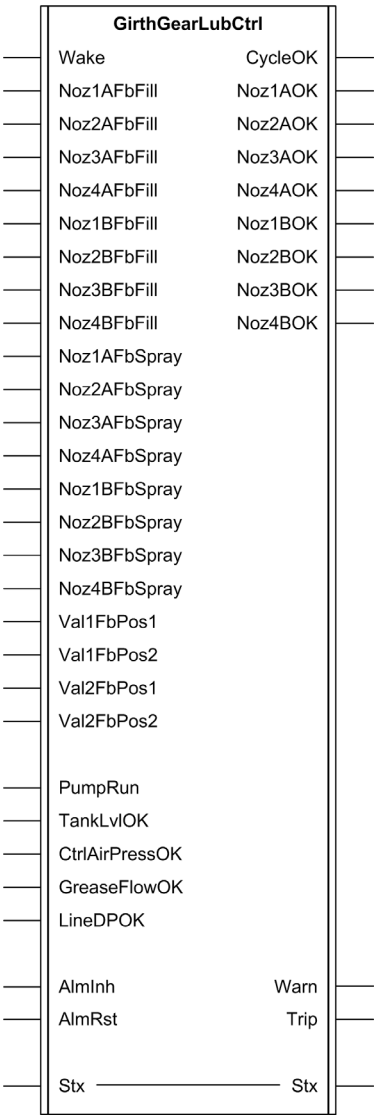
If the synchronization of nozzle feedback with valve position is lost for more than a pre-defined time (Stx.TFbSpray, Stx.TFbFill), an alarm is generated. The block also supports bypass of the nozzles in case of temporary site requirements. Alarms are not generated for the bypassed nozzles.

The block also checks for the inoperable conditions (tank level low, control air pressure low, grease flow low, line differential pressure low) of the girth gear lubrication system and generates relevant alarms. The block also supports bypass of the inoperable conditions in case of temporary site requirements. Alarms are not generated for the relevant inoperable conditions.

The alerts and alarms are indicated in the SCADA. The details of the active alarms are shown in the **Alarm** tab of the popup. Based on the user configured conditions, the alarms for this block can be inhibited by connecting the logic to the input pin AlmInh. The trip alarms can be reset by the input pin AlmRst, which is connected from GrpCtrl block or by the operator from popup.

Representation in FBD

The following figure shows the FBD representation of the `GirthGearLubCtrl` block.



Parameter Description

The following table describes the input parameters of the GirthGearLubCtrl block:

Parameter	Type	Description	Details
Wake	BOOL	Group wake-up	Group wake-up and request for availability check: • Wake = 1: Group is active. • Wake = 0: Group is in Sleep mode. This input is connected from output GWake of GrpCtrl block.
NozxAFbFill (x: 1...4)	BOOL	Fill feedback of nozzle xA (x: 1...4)	• NozxAFbFill (x: 1...4) = 1: It is active. • NozxAFbFill (x: 1...4) = 0: It is not active.
NozxBFbFill (x: 1...4)	BOOL	Fill feedback of nozzle xB (x: 1...4)	• NozxBFbFill (x: 1...4) = 1: It is active. • NozxBFbFill (x: 1...4) = 0: It is not active.
NozxAFbSpray (x: 1...4)	BOOL	Spray feedback of nozzle xA (x: 1...4)	• NozxAFbSpray (x: 1...4) = 1: It is active. • NozxAFbSpray (x: 1...4) = 0: It is not active.
Val1FbPos1	BOOL	Position 1 feedback of valve 1	• Val1FbPos1 = 1: It is active. • Val1FbPos1 = 0: It is not active.
Val2FbPos2	BOOL	Position 2 feedback of valve 1	• Val2FbPos2 = 1: It is active. • Val2FbPos2 = 0: It is not active.
Val2FbPos1	BOOL	Position 1 feedback of valve 2	• Val2FbPos1 = 1: It is active. • Val2FbPos1 = 0: It is not active.
Val2FbPos2	BOOL	Position 2 feedback of valve 2	• Val2FbPos2 = 1: It is active. • Val2FbPos2 = 0: It is not active.
PumpRun	BOOL	Pump run	• PumpRun = 1: Pump run status is active. • PumpRun = 0: Pump run status is not active.
TankLvlOK	BOOL	Tank level OK	• TankLvlOK = 1: Tank level condition is operable. • TankLvlOK = 0: Tank level condition is inoperable and an alarm is generated.

Parameter	Type	Description	Details
CtrlAirPressOK	BOOL	Control air pressure OK	• CtrlAirPressOK = 1: Control air pressure condition is operable. • CtrlAirPressOK = 0: Control air pressure condition is inoperable and an alarm is generated.
GreaseFlowOK	BOOL	Grease flow OK	• GreaseFlowOK = 1: Grease flow condition is operable. • GreaseFlowOK = 0:Grease flow condition is inoperable and an alarm is generated.
LineDPOK	BOOL	Line differential pressure OK	• LineDPOK = 1:Line differential pressure condition is operable. • LineDPOK = 0: Line differential pressure condition is inoperable and an alarm is generated.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

The AlmInh parameter to inhibit alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the GirthGearLubCtrl block:

Parameter	Type	Description	Details
AlmInh	BOOL	Alarm inhibition	• AlmInh = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • AlmInh = 0: Alarms are not inhibited.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

To reset, all generated alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the GirthGearLubCtrl block:

Parameter	Type	Description	Details
AlmRst	BOOL	Reset	• AlmRst = 1: Resets all the alarms generated by the block. • AlmRst = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from GRst pin of GrpCtrl block and resets the trip alarm.

The following table describes the output parameters of the GirthGearLubCtrl block:

Parameter	Type	Description	Details
CycleOK	BOOL	Cycle OK status	• CycleOK = 1: Girth gear lubrication cycle is OK. • CycleOK = 0: Girth gear lubrication cycle is not OK.
NozxAOK (x: 1...4)	BOOL	Noz xA (x: 1...4) OK status	• NozxAOK (x: 1...4) = 1: Nozzle xA (x: 1...4) is OK. • NozxAOK (x: 1...4) = 0: Nozzle xA (x: 1...4) is not OK.
NozxBOK	BOOL	Noz xB (x: 1...4) OK status	• NozxBOK (x: 1..4) = 1: Nozzle xB (x:1...4) is OK. • Noz1BOK (x: 1...4) = 0: Nozzle xB (x: 1...4) is not OK.
Warn	BOOL	Alert	• Warn = 1: The girth gear lubrication is operational with no active alert present. • Warn = 0: The girth gear lubrication is operational with an active alert present.
Trip	BOOL	Trip	• Trip = 1: The girth gear lubrication is operational with no active alarm present. • Trip = 0: The girth gear lubrication is operational with an active alarm present.

The following table describes the input/output parameter of the `GirthGear-LubCtrl` block:

Parameter	Type	Description
`Stx`	`GirthGearLubCtrl_St`	Status and control of the `GirthGearLubCtrl` block. Refer to the Structure to interface with the HMI *(see page 528)*.

`GirthGearLubCtrl` - Representation as SCoD in UAG

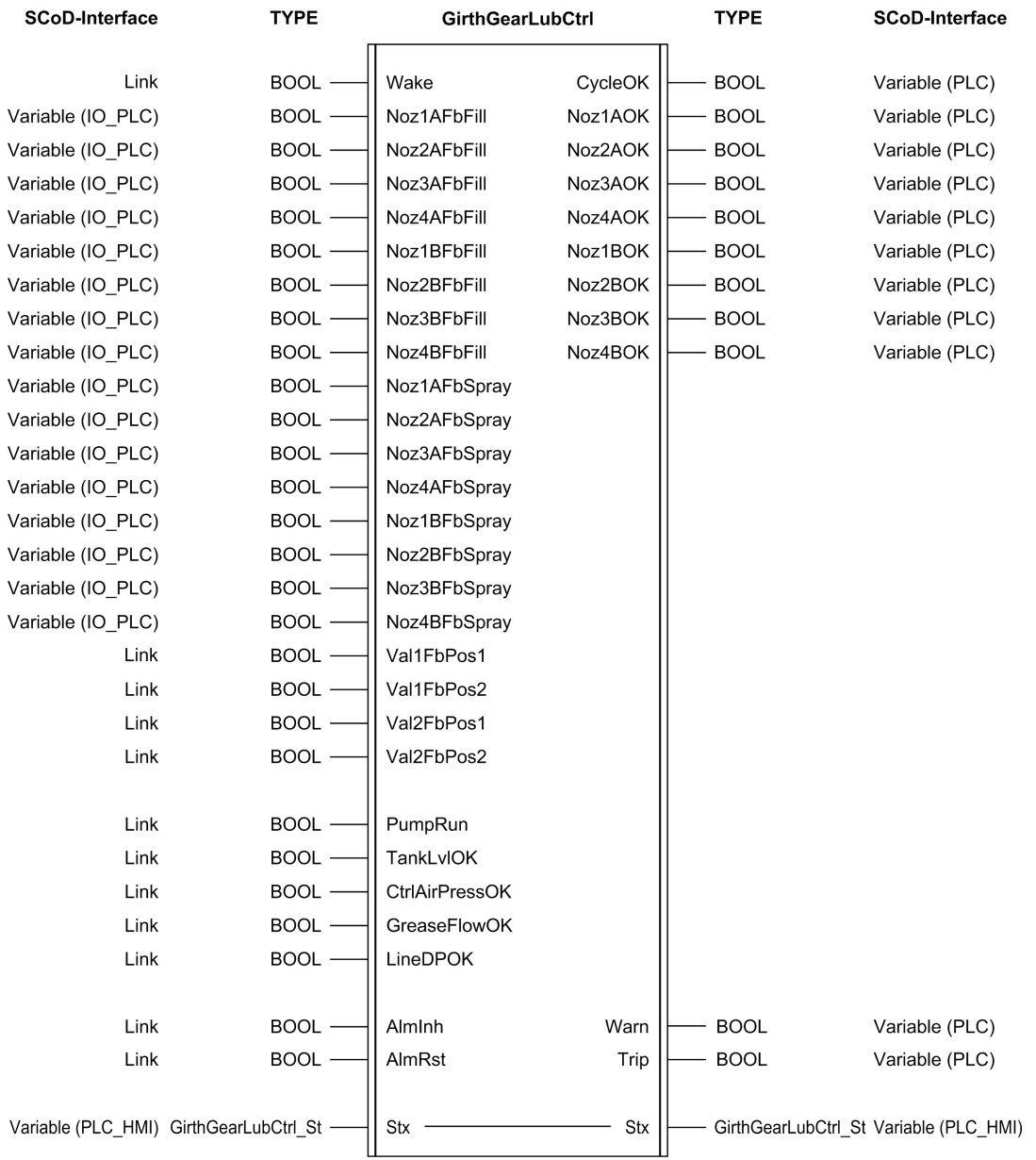
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the `Stx` variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type `HMI` is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the `GirthGearLubCtrl` block.



SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Display format (time)	`Stx.TFbSpray`	Not available	**Configuration**	1	Spray feedback time
Feedback Fill Delay (min)	`Stx.TFbFill`	5	**Configuration**	2	Fill feedback time
Feedback Spray Delay (min)	`Stx.TFbSpray`	5	**Configuration**	8	Spray feedback time

GirthGearLubCtrl - Representation in the HMI

Girth Gear Lubrication Control (Genie)

The GirthGearLubCtrl block is represented in the HMI by the following genie.

Genie Elements

Symbol	Meaning
GG Lub. Control	Opens the GirthGearLubCtrl popup.
Cem_GGLub	The instance name of the GirthGearLubCtrl block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

Item	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Animation Description of the Genie

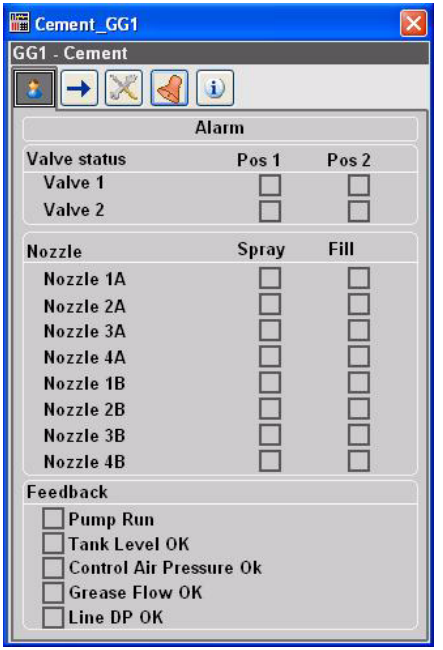
Symbol	Animation	Meaning
GG Lub. Control	Line color: Flashing yellow	Alert alarm for the GirthGearLubCtrl block is active.
GG Lub. Control	Line color: Steady yellow	Alert alarm for the GirthGearLubCtrl block is acknowledged.
GG Lub. Control	Line color: Flashing red	Trip alarm for the GirthGearLubCtrl block is active.
GG Lub. Control	Line color: Steady red	Trip alarm for the GirthGearLubCtrl block is acknowledged.

Popup

The `GirthGearLubCtrl` block popup has the following tabs:

- **Operator** tab
- **Extended** tab
- **Configuration** tab
- **Alarm** tab
- **Information** tab

Operator Tab



This tab has 4 sub-sections:

- **Status**
- **Valve status**
- **Nozzle**
- **Feedback**

The sub-section **Status** displays the current status.

The possible displays are listed as follows:

- **Sleeping**
- **Awake**
- **Cycle Ok**
- **Alarm**

The following table describes the sub-section **Valve status**:

Item	Animation	Description
Valve 1	Steady green	**Valve 1** position feedback is active.
	Steady gray	**Valve 1** position feedback is not active.
Valve 2	Steady green	**Valve 2** position feedback is active.
	Steady gray	**Valve 2** position feedback is not active.

The following table describes the sub-section **Nozzle**:

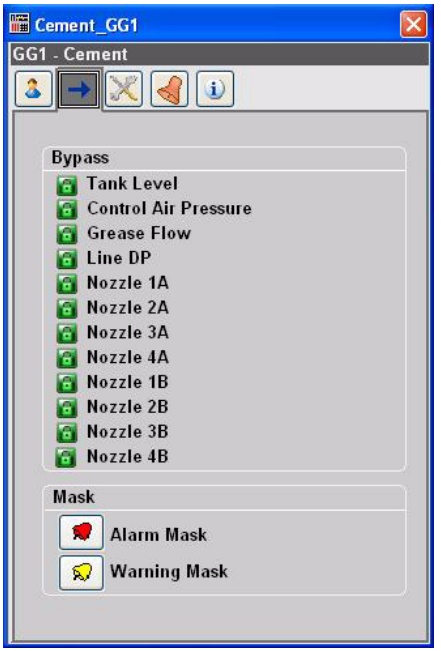
Item	Animation	Description
Nozzle1A	Steady green	Displays the spray feedback and fill feedback. **Nozzle 1A** is active.
	Steady gray	**Nozzle 1A** is not active.
Nozzle2A	Steady green	Displays the spray feedback and fill feedback. **Nozzle 2A** is active.
	Steady gray	**Nozzle 2A** is not active.
Nozzle3A	Steady green	Displays the spray feedback and fill feedback. **Nozzle 3A** is active.
	Steady gray	**Nozzle 3A** is not active.
Nozzle4A	Steady green	Displays the spray feedback and fill feedback. **Nozzle 4A** is active.
	Steady gray	**Nozzle 4A** is not active.
Nozzle 1B	Steady green	Displays the spray feedback and fill feedback. **Nozzle 1B** is active.
	Steady gray	**Nozzle 1B** is not active.
Nozzle 2B	Steady green	Displays the spray feedback and fill feedback. **Nozzle 2B** is active.
	Steady gray	**Nozzle 2B** is not active.
Nozzle 3B	Steady green	Displays the spray feedback and fill feedback. **Nozzle 3B** is active.
	Steady gray	**Nozzle 3B** is not active.
Nozzle 4B	Steady green	Displays the spray feedback and fill feedback. **Nozzle 4B** is active.
	Steady gray	**Nozzle 4B** is not active.

The following table describes the sub-section **Feedback**:

Item	Animation	Description
Pump Run	Steady green	Pump run status is active.
	Steady gray	Pump run status is not active.

Item	Animation	Description
Tank Level OK	Steady green	Tank level pressure condition is operable.
	Steady gray	Tank level pressure condition is inoperable.
Control Air Pressure OK	Steady green	Control air pressure condition is operable.
	Steady gray	Control air pressure condition is inoperable.
Grease Flow OK	Steady green	Grease flow condition is operable
	Steady gray	Grease flow condition is inoperable.
Line DP OK	Steady green	Line differential pressure condition is operable.
	Steady gray	Line differential pressure condition is inoperable.

Extended Tab



This tab has 2 sub-sections:

- **Bypass**
- **Mask**

The following table describes the sub-section **Bypass**:

Item	Description
Tank Level	Tank level monitoring is bypassed.
Control Air Pressure	Control air pressure monitoring is bypassed.
Grease Flow bypass	Grease flow monitoring is bypassed.
Nozzle 1A	Nozzle 1A monitoring/condition is bypassed.
Nozzle 2A	Nozzle 2A monitoring/condition is bypassed.
Nozzle 3A	Nozzle 3A monitoring/condition is bypassed.
Nozzle 4A	Nozzle 4A monitoring/condition is bypassed.
Nozzle 1B	Nozzle 1B monitoring/condition is bypassed.
Nozzle 2B	Nozzle 2B monitoring/condition is bypassed.
Nozzle 3B	Nozzle 3B monitoring/condition is bypassed.
Nozzle 4B	Nozzle 4B monitoring/condition is bypassed.

The following table describes the sub-section **Mask**:

Item	Description
Alarm Mask	Alarms are generated in alarm list, but auto acknowledged and hence color animation is not displayed.
Warning Mask	Alert alarms are generated in alarm list, but auto acknowledged and hence color animation is not displayed.

Configuration Tab

This tab has a sub-section called as **Delay**, which is described in the following table:

Item	Description
Spray Time	Displays spray feedback time.
Fill Time	Displays fill feedback time.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

GirthGearLubCtrl_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type GirthGearLubCtrl_St for the GirthGearLubCtrl block:

Element	Type	Description
TFbSpray	INT	Spray feedback time
TFbFill	INT	Fill feedback time
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Stw2	WORD	Status word 2 to interact with HMI
Stw3	WORD	Status word 3 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI
Alw2	WORD	Alarm word 2 to interact with HMI

Bit description for the Ctw element:

Bit	Description
0	Line differential pressure bypass
1	Control air pressure bypass
2	Tank level bypass
3	Grease flow bypass
4...6	Not used
7	Alarm reset
8...13	Not used
14	Alarm mask
15	Alert mask

Bit description for the Stw1 element:

Bit	Description
0	Wake
1	Start interlock
2	Safety interlock
3	Sequential interlock
4	Process interlock
5	Line differential pressure

Bit	Description
6	Control air pressure
7	Tank level
8	Grease flow
9...13	Not used
14	Alarm inhibition
15	Not used

Bit description for the `Stw2` element:

Bit	Description
0	Group active
1	Cycle active
2	Interlock
3	Ready
4	Run
5...13	Not used
14	Alert
15	Trip

Bit description for the `Alw1` element:

Bit	Description
0	Safety interlock not available
1	Sequential interlock not available
2	Process interlock not available
3	Line differential pressure inoperable
4	Control air pressure inoperable
5	Tank level inoperable
6	Grease flow inoperable
7	Run feedback forward inoperable when motor is running
8	Run feedback forward inoperable when motor is not running
9...10	Not used
11	Process feedback inoperable when motor is running
12	Run feedback inoperable when motor is not running
13...14	Not used
15	Trip

Generic Process

11

Overview

The following chapter describes the generic process objects.

⚠ WARNING

MISAPPLICATION OF MODULES

The modules in this section do not reflect any specific installation. Before adopting these modules for use in a specific application, the engineer must:

- conduct a safety analysis for the application and equipment installed
- verify that all modules are appropriate for the equipment or function in the installation
- supply appropriate parameters, particularly for limits
- check that all sensors and actuators are compatible with the modules selected
- thoroughly test all functions during verification and commissioning
- provide independent paths for critical control functions (emergency stop, over-limit conditions etc.) according to the safety analysis and applicable codes and regulations.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

What's in this Chapter?

This chapter contains the following sections:

Section	Topic	Page
11.1	`SPBool`: Set Point Bool	533
11.2	`SPInt`: Set Point Integer	539
11.3	`SPReal`: Set Point Real	547
11.4	`RatioCtrl`: Ratio Controller	555
11.5	`Totaliser`: Totaliser	575
11.6	`EgySchE`: Energy Scheduler Event Based	583

Section	Topic	Page
11.7	EgySchT: Energy Scheduler Time Based	594
11.8	ILCK: Interlock	605
11.9	PIDB: Proportional, Integral and Derivative Controller Block	610

11.1 SPBool: Set Point Bool

Overview

This section describes the SPBool block.

What's in this Section?

This section contains the following topics:

Topic	Page
SPBool - Description	534
SPBool - Representation as SCoD in UAG	536
SPBool - Representation in the HMI	537
SPBool_St Derived Data Type	538

SPBool - Description

General

The SPBool block is implemented as follows:

Software	Implemented as
Unity Pro	DFB SPBool
Vijeo Citect	Genie SPBool
UAG	SCoD SPBool

Function Description

The SPBool block is used for the applications, where the operator enters the set points of Bool format from the HMI. This block accepts the input set point in the Bool format.

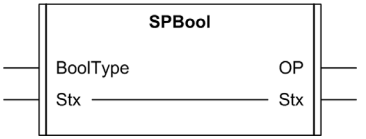
You can select 2 input types as described in the following table:

Input Type	Description
Latch/Unlatch	The block accepts input set point in Bool and latches the output. If the output is unlatched, the block receives the Bool input set point again.
Pulse	The block accepts the set point in Bool and gives a pulse output of fixed length (500ms).

The input type whether **Latch/Unlatch** or **Pulse** type can be configured during the instantiation of the block.

Representation in FBD

The following figure shows the FBD representation of the SPBool block.



Parameter Description

The following table describes the input parameter of the SPBool block:

Parameter	Type	Description	Details
BoolType	BOOL	Type	• BoolType = 1: Pulse mode is selected. • BoolType = 0: Latch/Unlatch mode is selected.

The following table describes the output parameter of the SPBool block:

Parameter	Type	Description	Details
OP	BOOL	Output	Boolean set point output.

The following table describes the input/output parameter of the SPBool block:

Parameter	Type	Description
Stx	SPBool_St	Status and control of the SPBool set point block. Refer to the Structure to interface with the HMI. *(see page 538)*

SPBool - Representation as SCoD in UAG

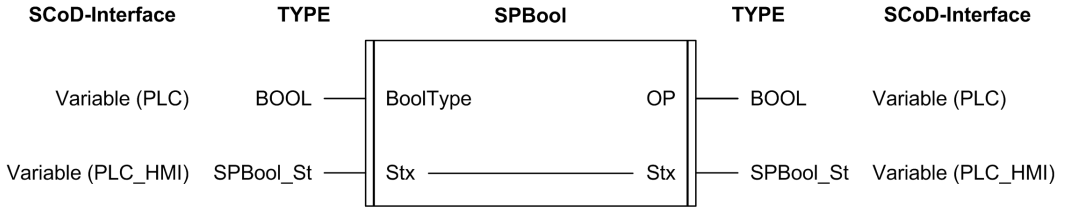
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the Stx variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type HMI is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the SPBool block.



SPBool - Representation in the HMI

Set Point Bool (Genie)

The SPBool block is represented in the HMI by the following genie:

Genie Elements

Symbol	Meaning
	This button is visible when the **Latch/Unlatch** type is selected during instantiation or when the boolean set point output is unlatched. On clicking this button, a boolean set point output is latched.
	This button is visible when the **Latch/Unlatch** type is selected and the boolean set point output is latched. On clicking this button, a boolean set point output is unlatched.
	This button is visible when the **Pulse** type is selected. On clicking this button, a pulse of defined length is given with a boolean set point output.
	The instance name of the SPBool block is visible by default. You can hide/unhide it by clicking the **Hide/Unhide** button in the template.

Popup

The boolean set point block does not have any popup.

SPBool_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type SPBool_St for the SPBool block:

Element	Type	Description
SP	BOOL	Set point from HMI
OP	BOOL	Output
Dummy1	BOOL	Not used
Dummy2	BOOL	Not used

11.2 SPInt: Set Point Integer

Overview

This section describes the SPInt block.

What's in this Section?

This section contains the following topics:

Topic	Page
SPInt - Description	540
SPInt - Representation as SCoD in UAG	542
SPInt - Representation in the HMI	543
SPInt_St Derived Data Type	546

SPInt - Description

General

The SPInt block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB SPInt
Vijeo Citect	Genie SPInt
UAG	SCoD SPInt

Function Description

This block is used for the applications, where the operator enters the set points of integer types from HMI. This block accepts the input set point in the Int (Integer) format (negative and positive integer values).

You can select 2 input types as described in the following table:

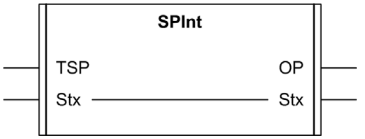
Input Type	Description
Free Input	The input set point can be any integer value that an operator enters from HMI.
Limited Input	The input set point has to be within the limits, set point high and set point low configured by the user. The operator can enter the value only within the limits.

The input type, high and low values can be configured during instantiation of the block and in the popup as well.

In addition, you can configure the delay in changing the set point value from old value to new value through HMI.

Representation in FBD

The following figure shows the FBD representation of the SPInt block.



Parameter Description

The following table describes the input parameter of the SPInt block:

Parameter	Type	Description	Details
TSP	TIME	Set point delay	Delay time to change the set point SP from old value to new value.

The following table describes the output parameter of the SPInt block:

Parameter	Type	Description
OP	INT	Integer set point output

The following table describes the input/output parameter of the SPInt block:

Parameter	Type	Description
Stx	SPInt_St	Status and control of the SPInt block. Refer to the Structure to interface with the HMI *(see page 546)*.

SPInt - Representation as SCoD in UAG

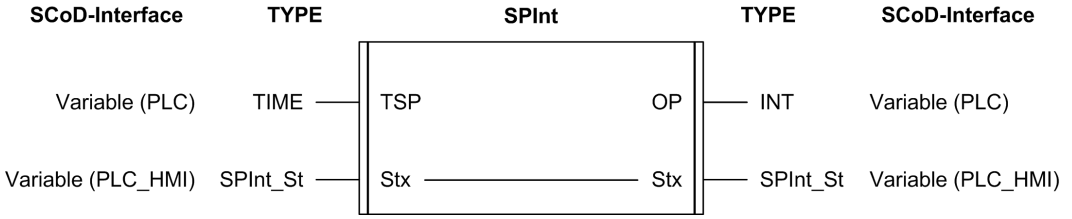
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the Stx variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type HMI is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the SPInt block.



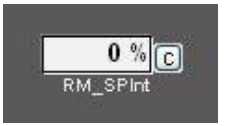
SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Display format	Stx.SP	0	**Configuration**	2	Display format
Measurement unit	Stx.SP	0	**Configuration**	7	Measurement unit
Set Point	Stx.SP	0	**Configuration**	1	Set point
Output High Limit	Stx.SPH	100	**Limit**	7	Output high limit
Output Low Limit	Stx.SPL	0	**Limit**	1	Output low limit

SPInt - Representation in the HMI

Set Point Integer (Genie)

The SPInt block is represented in the HMI by the following genie:



Genie Elements

Symbol	Meaning
	Indicates the integer set point output of the block. On selecting this text box, the operator can enter the new real set point.
	Opens the integer set point popup.
	The instance name of the SPInt block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

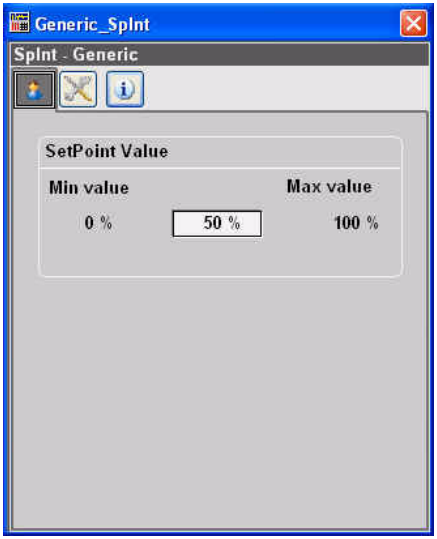
Item	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Popup

The SPInt block popup has the following tabs:

- **Operator** tab
- **Configuration** tab
- **Information** tab

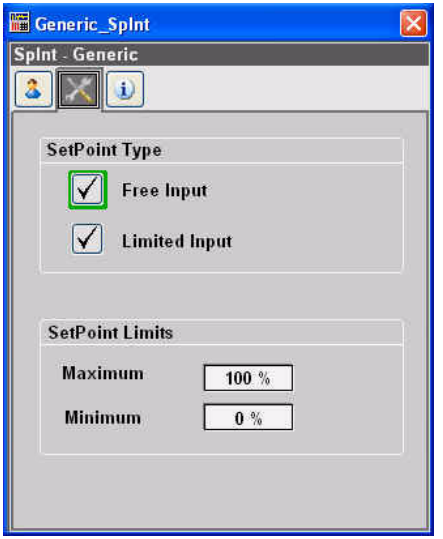
Operator Tab



The following table describes the sub-section **SetPoint Value**:

Symbol	Description
	If the input type selected is free, then you can enter any integer set point. If the input type selected is limited, then you can enter integer set point within maximum and minimum value.
	Displays the configured minimum value.
	Displays the configured maximum value.

Configuration Tab



The following tab has two sub-sections:

- **SetPoint Type**
- **SetPoint Limits**

The following table describes the sub-section **SetPoint Type**:

Button	Description
Free Input	On clicking this button, the set point type is configured as **Free Input**. A green background to this button is displayed indicating the selection.
Limited Input	On clicking this button, the set point type is configured as **Limited Input**. A green background to this button is displayed indicating the selection.

The following table describes the sub-section **SetPoint Limits**:

Button	Description
Maximum	User can configure the maximum set point value.
Minimum	User can configure the minimum set point value.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

SPInt_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type SPInt_St for the SPInt block:

Element	Type	Description
SP	INT	Set point from HMI
OP	INT	Output
SPH	INT	Set point high limit
SPL	INT	Set point low limit
Dummy	INT	Not used
Typ	BOOL	Selection of free or limited input mode

11.3 SPReal: Set Point Real

Overview

This section describes the SPReal block.

What's in this Section?

This section contains the following topics:

Topic	Page
SPReal - Description	548
SPReal - Representation as SCoD in UAG	550
SPReal - Representation in the HMI	551
SPReal_St Derived Data Type	554

SPReal - Description

General

The SPReal block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB SPReal
Vijeo Citect	Genie SPReal
UAG	SCoD SPReal

Function Description

The SPReal block is used for the applications, where the operator enters the set points of real type from HMI. This block accepts the input set point in the Real format.

You can select 2 input types as described in the following table:

Input Type	Description
Free Input	The input set point can be any real value that an operator enters from HMI.
Limited Input	The input set point has to be within the limits output high and output low. The operator can enter the value only within the limits.

The input type, output high and low values can be configured during instantiation of the block and in the popup as well. Delay time is not configurable through popup.

Representation in FBD

The following figure shows the FBD representation of the SPReal block.

Parameter Description

The following table describes the input parameter of the SPReal block:

Parameter	Type	Description	Details
TSP	TIME	Set point delay	Delay time to change the set point SP from old value to new value.

The following table describes the output parameter of the SPReal block

Parameter	Type	Description
OP	REAL	Real set point output

The following table describes the input/output parameter of the SPReal block:

Parameter	Type	Description
Stx	SPReal_St	Status and control of the SPReal set point block. Refer to the Structure to interface with the HMI. *(see page 554)*

SPReal - Representation as SCoD in UAG

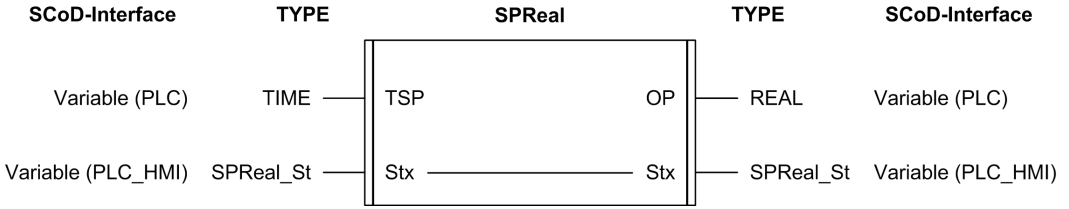
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the Stx variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type HMI is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface forthe SPReal block.



SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Display format	Stx.SP	0.0	**Configuration**	2	Display format
Measurement unit	Stx.SP	0.0	**Configuration**	7	Measurement unit
Set Point	Stx.SP	0.0	**Configuration**	1	Set point
Output High Limit	Stx.SPH	100.0	**Limit**	7	Output high limit
Output Low Limit	Stx.SPL	0.0	**Limit**	1	Output low limit

SPReal - Representation in the HMI

Set Point Real (Genie)

The SPReal block is represented in the HMI by the following genie:

Genie Elements

Symbol	Meaning
	Displays the real set point output of the block. On clicking this text box, the operator can enter the new integer set point value.
	Opens the real set point block popup.
	The instance name of the SPReal block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

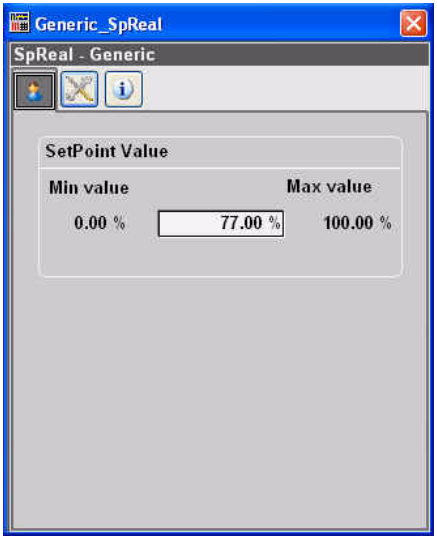
Item	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Popup

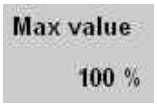
The SPReal block popup has the following tabs:

- **Operator** tab
- **Configuration** tab
- **Information** tab

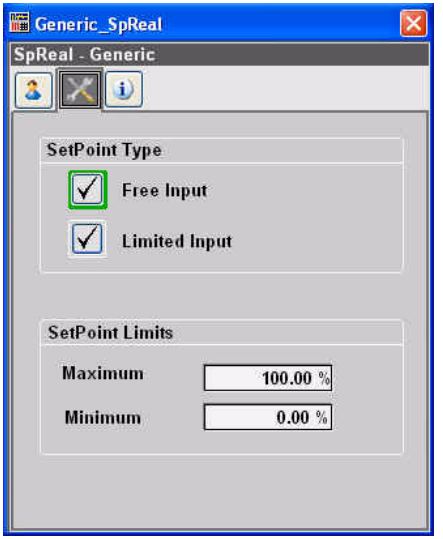
Operator Tab



The following table describes the sub-section **Set point Value**:

Symbol	Description
	If the input type selected is free, then the user can enter any real set point. If the input type selected is limited, then the user can enter integer set point within maximum and minimum value.
	If the input type selected is limited, then the configured minimum value is displayed.
	If the input type selected is limited, then the configured maximum value is displayed.

Configuration Tab



The follwoing tab has two sub-sections:

- **SetPoint Type**
- **SetPoint Limits**

The following table describes the sub-section **SetPoint Type**:

Button	Description
Free Input	On selecting this button, the set point type is configured as **Free Input**. A green background to this button is displayed indicating the selection.
Limited Input	On selecting this button, the set point type is configured as **Limited Input**. A green background to this button is displayed indicating the selection.

The following table describes the sub-section **SePoint Limits**:

Button	Description
Maximum	User can configure the maximum set point value.
Minimum	User can configure the minimum set point value.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

SPReal_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type SPReal_St for the SPReal block:

Element	Type	Description
SP	REAL	Set point from HMI
OP	REAL	Output
SPH	REAL	Set point high limit
SPL	REAL	Set point low limit
Typ	BOOL	Selection of free or limited input mode
Dummy1	BOOL	Not used
Dummy2	BOOL	Not used

11.4 RatioCtrl: Ratio Controller

Overview

This section describes the RatioCtrl block.

What's in this Section?

This section contains the following topics:

Topic	Page
RatioCtrl - Description	556
RatioCtrl - Representation as SCoD in UAG	563
RatioCtrl - Representation in the HMI	566
RatioCtrl_St Derived Data Type	572

RatioCtrl - Description

General

The RatioCtrl block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB RatioCtrl
Vijeo Citect	Genie RatioCtrl
UAG	SCoD RatioCtrl

Function Description

This RatioCtrl block is used to control the ratio of ingredients of a continuous process.

The main application example is recipe management of the mill feed. This block can support up to 6 ingredients and their ratios. This block generates 6 set points to individual actuators that indirectly maintains a ratio between 6 different ingredients.

This block supports 3 modes of operation:

- **Auto mode**
- **Manual mode**
- **External mode**

The block goes to Default mode configured (DMode) during initialization. The block will be forced to Auto or External mode if the respective mode selection input pin (AutoSel or ExtSel) is HIGH.

If the mode selection input pin (AutoSel or ExtSel) is LOW, operator can change mode of the RatioCtrl block from the popup of the block.

Auto Mode Operation

The total set point is fed to the block through input pin (ExtTotalSP) or by the operator from HMI (Stx.ManTotalSP) based on the external set point enable/disable. The individual set point ratios ARatiox (x: 1..6) are entered in percentage by the operator from the popup. When the command **UpdateRatio** is given by the operator from HMI and OPxEn (x: 1...6) is enabled, then the ratio is updated to the final ratio of the outputs Ratiox (x: 1...6). Set point SPx (x: 1...6) is calculated by the formulae SPx = TotalSP * Ratiox/100. When the ratio or the total set point is changed, the set point outputs are adjusted accordingly to maintain the ratio. The generated set points are restricted to the limits SPxH and SPxL (x: 1...6) entered by the operator from HMI. If the total ratio is not equal to 100% for more than pre-defined time (Stx.TRatioError), then an alert is generated.

Manual Mode Operation

When the block is in Manual mode, the manual set point `Stx.ManSPx` (x: 1...6) is entered by the operator in the popup, and is generated as the output set point `SPx` (x: 1...6).

Manual set point entry from HMI is restricted to the limits `SPxH` and `SPxL` (x: 1...6).

Cascade/External Mode Operation

The total set point is fed to the block through input pin (`ExtTotalSP`) or by the operator from HMI (`Stx.ManTotalSP`) based on the external set point enable or disable.

The individual set point ratios are connected through the input pins `ExtRatiox` (x: 1...6). If `OPxEn` (x: 1...6) is enabled, then the ratio is updated to the final ratio of the outputs `Ratiox` (x: 1...6). Set point `SPx` (x: 1...6) is calculated by the formuale `SPx = TotalSP * Ratiox/100`.

When the ratio or the total set point is changed, the set point outputs are adjusted accordingly to maintain the ratio. The generated set points are restricted to the limits `SPxH` and `SPxL` (x: 1...6) entered by the operator from HMI.

If the total ratio is not equal to 100% for more than pre-defined time (`Stx.TRatioError`), then an alert is generated.

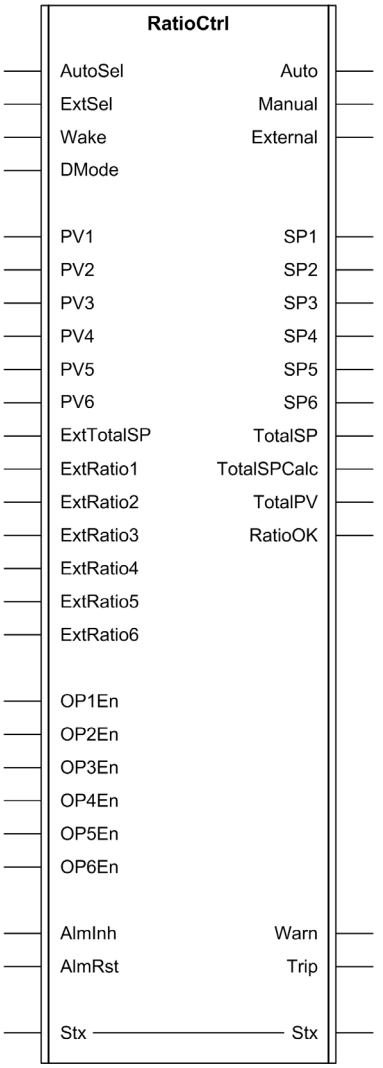
Limits and Conditions

The `RatioCtrl` block continuously monitors the process feedback value `PVx` (x: 1...6). The value of the `PVx` (x: 1...6) and total process values are displayed in the popup (**PV, Total PV)** of the `RatioCtrl` block. If the total process value deviates from `TotalSP` beyond limit (`Dev`) for a longer time (`TDev`), then an alert is generated.

In the Cascade/External mode or Auto mode, if the input and output set points are not equal, then an alarm is generated. The alert and alarms are indicated in the SCADA. The details of the active alarms are shown in the **Alarm** tab of the popup.

Representation in FBD

The following figure shows the FBD representation of the `RatioCtrl` block.



Parameter Description

The following table describes the input parameters of the RatioCtrl block:

Parameter	Type	Description	Details
AutoSel	BOOL	Auto mode selection	• AutoSel = 1: Block is forced to Auto mode. • AutoSel = 0: Block is set free for operator to select the mode from HMI. Set points are generated based on the ratio fixed by operator from HMI. **NOTE:** Mode selection option in HMI is grayed out.
ExtSel	BOOL	External mode selection	• ExtSel = 1: Block is forced to External mode. • ExtSel = 0: Block is set free for operator to select the mode from HMI. Set points are generated based on the ratio inputs that comes from other blocks or any other system, for example, Quality control systems). **NOTE:** Mode selection option in HMI is grayed out. This will be connected from other blocks or systems (Expert control systems).
Wake	BOOL	Group wake-up	Group wake-up and request for availability check: • Wake = 1: Group is active. • Wake = 0: Group is in Sleep mode. This input is connected from output GWake of GrpCtrl block.
DMode	INT	Default mode	The block can be set to the default mode of operation during initialization. Selection of Operation mode: • Dmode = 1: Auto mode • Dmode = 2: Manual mode • Dmode = 3: External mode
PVx (x: 1...6)	REAL	Process value feedback x (x: 1...6)	If SPx - PVx (x: 1...6) > MaxDev for more than time TMaxDev, then an alarm is generated.

Parameter	Type	Description	Details
ExtTotalSP	REAL	External total set point	• Application is in External and Auto modes only if the ExtTotalSPEnable is selected from popup. • In External mode, it is divided into individual set points based on the ratio input pins. In Auto mode, it is divided into individual set points based on the ratio entered by operator from HMI. • Shall be connected from other blocks, for example, systems (Expert control systems) or PID Controller.
ExtRatiox x (x: 1...6)	REAL	External ratio for set points x (x:1...6)	Applicable in External modes only.
OPxEn (x: 1...6)	BOOL	Output x (x: 1...6) enable	• OPxEn = 1: Set point x (x: 1...6) is generated. • OPxEn = 0: Set point x (x: 1...6) is not generated.

⚠ CAUTION

UNINTENDED MACHINE OR PROCESS BEHAVIOR

Be sure of the state of security of your machine or process environment.

The AlmInh parameter to inhibit alarms must be used very carefully.

Failure to follow these instructions can result in injury or equipment damage.

The following table describes the input parameter of the RatioCtrl block:

Parameter	Type	Description	Details
AlmInh	BOOL	Alarm inhibition	• AlmInh = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • AlmInh = 0: Alarms are not inhibited.

⚠ CAUTION
UNINTENDED MACHINE OR PROCESS BEHAVIOR Be sure of the state of security of your machine or process environment. To reset, all generated alarms must be used very carefully. **Failure to follow these instructions can result in injury or equipment damage.**

The following table describes the input parameter of the `RatioCtrl` block:

Parameter	Type	Description	Details
`AlmRst`	`BOOL`	Reset	• `AlmRst` = 1: Resets all the alarms generated by the block. • `AlmRst` = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from `GRst` pin of `GrpCtrl` block and resets the trip alarm.

The following table describes the output parameter of the `RatioCtrl` block:

Output	Data Type	Description	Details
`Auto`	`BOOL`	Auto mode selected	• `Auto` = 1: Auto mode is activated and set points are generated based on the ratio fixed by operator from popup. • `Auto` = 0: Auto mode is not activated.
`Manual`	`BOOL`	Manual mode selected	• `Manual` = 1: Manual mode is activated and set points are directly entered by the operator from popup. • `Manual` = 0: Manual mode is not activated.
`External`	`BOOL`	External mode selected	• `External` = 1: External mode is activated and set points are generated based on ratio inputs that are coming from blocks or any other systems, for example, Quality control systems. • `External` = 0: External mode is not activated.

Output	Data Type	Description	Details
SPx x (x: 1...6)	REAL	Output set point x (x:1...6)	Applicable only if OPx (x: 1...6) = 1. • In External mode, SPx (x:1...6) = TotalSP * (ExtRatiox (x:1...6)/100 • In Auto mode, SPx (x: 1...6) = TotalSP * AutoSPx (x: 1...6)/100 • In Manual mode, SPx (x: 1...6) = ManSPx (x: 1...6) • Always SPxL (x: 1...6) < SPx (x: 1...6) < SPHx (x: 1...6) is maintained.
TotalSP	REAL	Total set point	This is the actual set point given to the ratio controller to generate outputs: • If ExtTotalSPEnable = 1, TotalSP = ExtTotalSP • If ExtTotalSPEnable = 0, TotalSP = Stx.ManTotalSP.
TotalSPcalc	REAL	Total set point	Calculated the total set point without considering the limits.
TotalPV	REAL	Total process value	It is equal to the summation of generated process values PVx (x: 1... 6).
RatioOK	BOOL	RatioOK	If sum of ratios is 100%, then RatioOK = 1
Warn	BOOL	Alert	• Warn = 1: The block is operational with no active alert present. • Warn = 0: The block is operational with an active alert present.
Trip	REAL	Trip	• Trip = 1: The block is operational with no active alarm present. • Trip = 0: The block is operational with an active alarm present.

The following table describes the input/output parameter of the RatioCtrl block:

Parameter	Type	Description
Stx	RatioCtrl_St	Status and control of the RatioCtrl block. Refer to the Structure to interface with the HMI *(see page 572)*.

`RatioCtrl` - Representation as SCoD in UAG

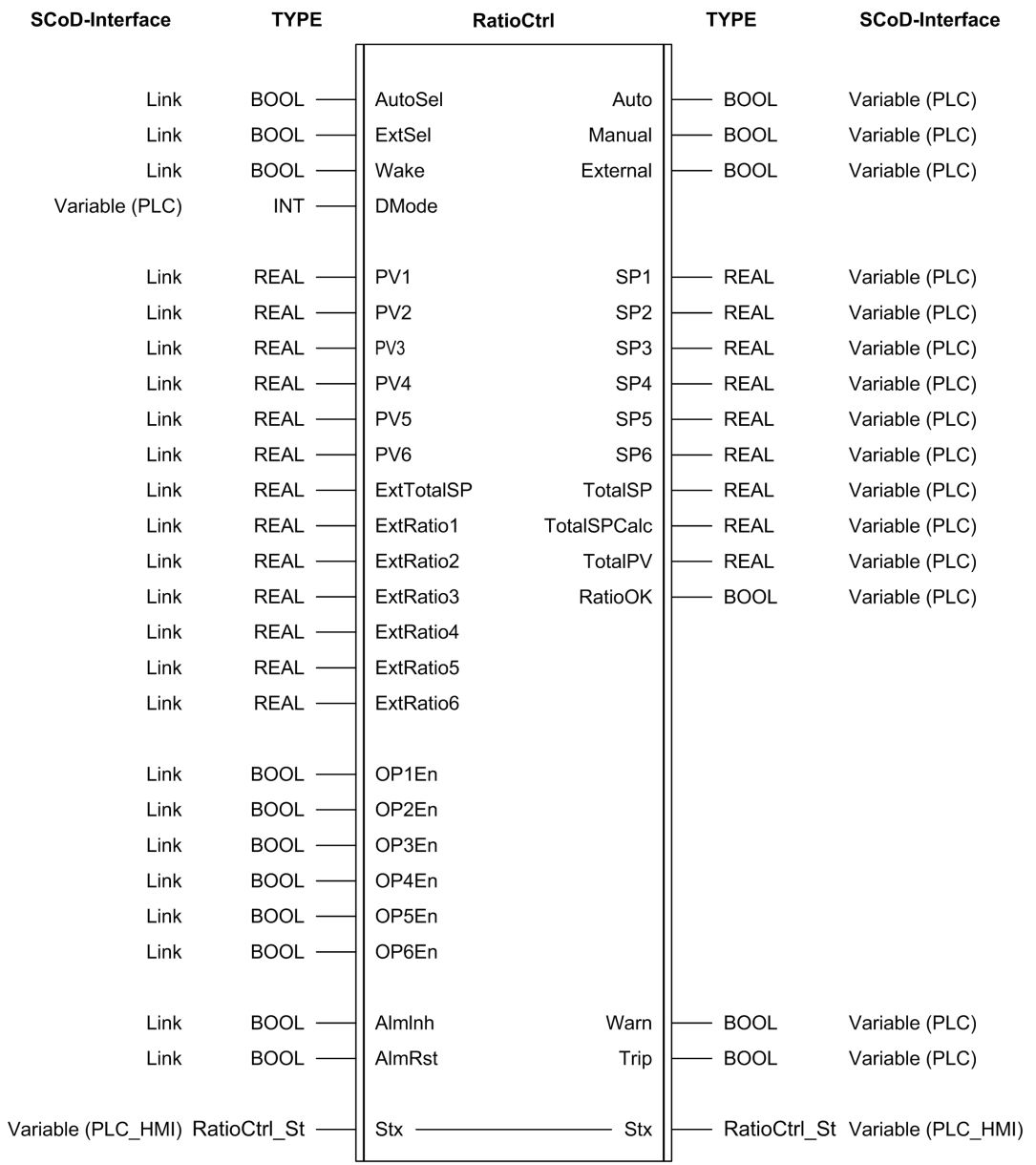
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the `Stx` variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type `HMI` is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the `RatioCtrl` block.



SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Default Mode	DMode	1-Auto	**Configuration**	2	Default mode
Display Format (SP)	Stx.TotalSP	Not available	**Configuration**	7	Display format (SP)
Display format (time)	Stx.TMaxPVDev	0	**Configuration**	1	Display format (time)
Max deviation value	Stx.MaxPVDev	0.0	**Configuration**	3	Maximum process value deviation permissible
Max deviation delay	Stx.TMaxPVDev	0	**Configuration**	9	Time delay for maximum process value deviation error
Measurement Unit	Stx.TotalSP	Not available	**Configuration**	8	Total setpoint output
Ratio Error Delay	Stx.TRatioError	Not available	**Configuration**	4	Time delay for ratio error
Set point Error Delay	Stx.TSPError	0	**Configuration**	10	Time delay for set point error
High limit for SP1	Stx.SP1H	100.0	**Limit**	7	High limit for set point 1
High limit for SP2	Stx.SP2H	100.0	**Limit**	8	High limit for set point 2
High limit for SP3	Stx.SP3H	100.0	**Limit**	9	High limit for set point 3
High limit for SP4	Stx.SP4H	100.0	**Limit**	10	High limit for set point 4
High limit for SP5	Stx.SP5H	100.0	**Limit**	11	High limit for set point 5
High limit for SP6	Stx.SP6H	100.0	**Limit**	12	High limit for Set point 6
Low limit for SP1	Stx.SP1L	0.0	**Limit**	1	Low limit for set point 1
Low limit for SP2	Stx.SP2L	0.0	**Limit**	2	Low limit for set point 2
Low limit for SP3	Stx.SP3L	0.0	**Limit**	3	Low limit for set point 3
Low limit for SP4	Stx.SP4L	0.0	**Limit**	4	Low limit for set point 4
Low limit for SP5	Stx.SP5L	0.0	**Limit**	5	Low limit for set point 5
Low limit for SP6	Stx.SP6L	0.0	**Limit**	6	Low limit for Set point 6
Ratio 1 Name	Ratio1Desc	Not available	**Ratio Names**	1	Ratio 1 description
Ratio 2 Name	Ratio2Desc	Not available	**Ratio Names**	7	Ratio 2 description
Ratio 3 Name	Ratio3Desc	Not available	**Ratio Names**	2	Ratio 3 description
Ratio 4 Name	Ratio4Desc	Not available	**Ratio Names**	8	Ratio 4 description
Ratio 5 Name	Ratio5Desc	Not available	**Ratio Names**	3	Ratio 5 description
Ratio 6 Name	Ratio6Desc	Not available	**Ratio Names**	9	Ratio 6 description

`RatioCtrl` - Representation in the HMI

Ratio Controller (Genie)

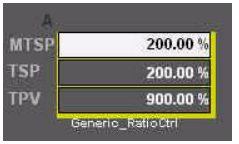
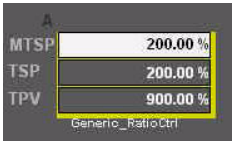
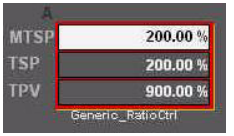
The `RatioCtrl` block is represented in the HMI by the following genie.

Genie Elements

Symbol	Meaning
	Opens the `RatioCtrl` popup.
	Displays the manual total set point value. User can enter the value through genie.
	Displays the total set point value.
	Displays the total process value.
	The instance name of the `RatioCtrl` block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

Item	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Animation Description of the Genie

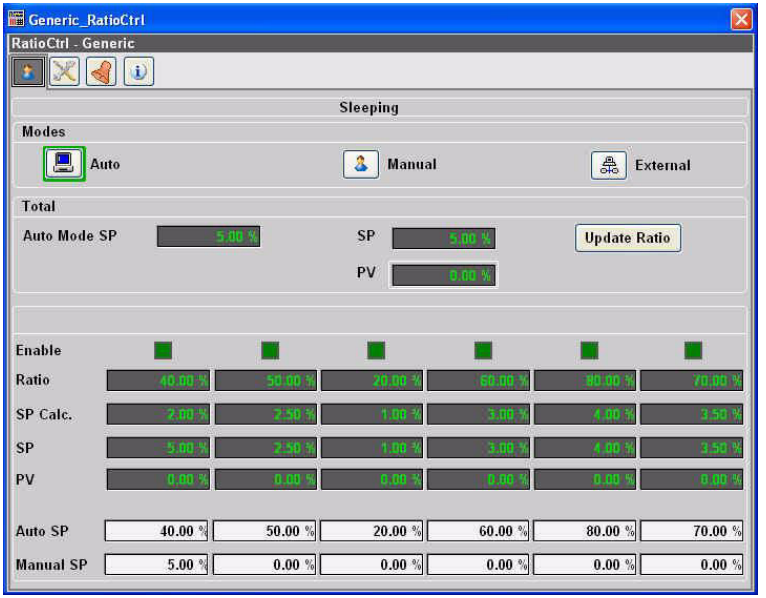
Symbol	Animation	Meaning
	Flashing yellow	Alert for the `RatioCtrl` is active.
	Steady yellow	Alert for the `RatioCtrl` is acknowldged.
	Flashing red	Alarm for the `RatioCtrl` is active.
	Steady red	Alarm for the `RatioCtrl` is acknowledged.

Popup

The `RatioCtrl` block popup has the following tabs:

- **Operator** tab
- **Configuration** tab
- **Alarm** tab
- **Information** tab

Operator Tab



This tab has 4 sub-sections:

- **Status**
- **Auto**
- **Manual**
- **External**

The **Status** sub-section displays the current status.

The possible displays are listed as follows:

- **Sleeping**
- **Awake**
- **Ratio Ok**
- **Warning**
- **Alarm**

The following table describes the sub-section **Modes**:

Item	Description
Auto	Switches to Auto operation mode to generate the set point output based on the ratio entered from the **ARatio** text fields.
Manual	Switches to Manual operation mode to generate the set point output based on the value set from the **Manual Set Point** text fields.
External	Switches to External operation mode to generate the set point output based on external set points.

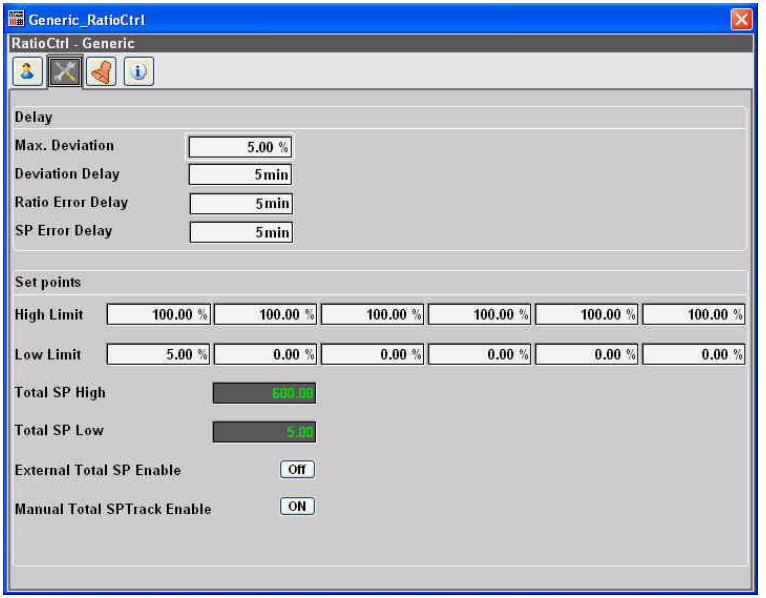
The following tabel describes the sub-section **Total**:

Item	Description
Auto Mode SP	Displays the Auto mode set point value.
SP	Displays the set point value.
PV	Displays the process value.
Update Ratio	Updates the ratio used for calculation of set points with the new values entered by the operator.

The following table describes the sub-section **R1…R6** (ratio numbers):

Item	Description
Enable	Displays the ratio output status: ● Steady green = enabled. ● Steady red = disabled.
Ratio	Displays the final ratio.
SP Calc.	Displays the set point calculation value.
SP	Displays the final set point.
PV	Displays the process value.
Auto SP	Enter the individual set point ratios for the Auto mode.
Manual SP	Enter the individual set point ratios for the Manual mode.

Configuration Tab



This tab has 2 sub-sections:

- **Delay**
- **Set points**

The following table describes the sub-section **Delay**:

Item	Description
Maximum Deviation	Displays the maximum deviation.
Deviation Delay	Displays the deviation delay.
Ratio Error Delay	Displays the ratio error time delay.
SP Error Delay	Displays the set point error time delay.

The following table describes the sub-section **Set points** (ratio numbers):

Item	Description
High Limit	Displays the high limit of the set point.
Low Limit	Displays the low limit of the set point.
Total SP High	Displays the total set point high value.

Item	Description
Total SP Low	Displays the total set point low value.
External Total SP Enable	Enable/Disable the external total set point.
Manual Total SPTrack Enable	Enable/Disable the manual total set point track.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Information Tab

For description about **Information** tab, refer to Commonly used Tabs *(see page 52)*.

RatioCtrl_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type RatioCtrl_St for the RatioCtrl block:

Element	Type	Description
ManTotalISP	REAL	Manual total set point
TotalSP	REAL	Total set point output
TotalSPH	REAL	Total set point high limit
TotalSPL	REAL	Total set point low limit
TotalSPcalc	REAL	Calculated total set point output
TotalPV	REAL	Total process value
AutoRatiox (x: 1...6)	REAL	Auto mode ratio set point x (x: 1...6)
Ratiox x (x: 1...6)	REAL	Final ratio x (x:1...6)
ManSPx (x:1...6)	REAL	Manual mode set to point x (x: 1...6)
SPxcalcx (x: 1...6)	REAL	Calculated set point x (x: 1...6)
SPx (x: 1...6)	REAL	Set point x (x: 1...6)
SPxH (x: 1...6)	REAL	Set point x (x: 1...6) high limit
SPxL (x: 1...6)	REAL	Set point x (x: 1...6) low limit
PVx (x: 1...6)	REAL	Process value x (x: 1...6)
MAXPVDev	REAL	Maximum process value deviation permissable
TMAXPVDev	INT	Time delay for maximum process value deviation error
TRatioError	INT	Time delay for ratio error
TSPError	INT	Time delay for set point error
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI
Dummy1	WORD	Not used
Dummy2	WORD	Not used

Bit description for the Ctw element:

Bit	Description
0	Auto
1	Manual
2	Not used

Bit	Description
3	External
4	Update ratio
5	External total set point enable
6	Manual total set point track enable
7	Alarm reset
8...13	Not used
14	Alarm masking
15	Alert masking

Bit description for the `Stw1` element:

Bit	Description
0	Wake
1	Auto mode selection
2	External mode selection
3	Output 1 enable
4	Output 2 enable
5	Output 3 enable
6	Output 4 enable
7	Output 5 enable
8	Output 6 enable
9	Auto
10	External
11	Manual
12	Ratio ok
13	Alert
14	Alarm inhibition
15	Trip

Bit description for the `Alw1` element:

Bit	Description
0	Set point 1 limit error
1	Set point 2 limit error
2	Set point 3 limit error
3	Set point 4 limit error

Bit	Description
4	Set point 5 limit error
5	Set point 6 limit error
6	Maximum process value 1 deviation
7	Maximum process value 2 deviation
8	Maximum process value 3 deviation
9	Maximum process value 4 deviation
10	Maximum process value 5 deviation
11	Maximum process value 6 deviation
12	Ratio error
13	More than one mode is selected in input pin
14	not used
15	Not used

11.5 Totaliser: Totaliser

Overview

This section describes the Totaliser block.

What's in this Section?

This section contains the following topics:

Topic	Page
Totaliser - Description	576
Totaliser - Representation as SCoD in UAG	578
Totaliser - Representation in the HMI	579
Totaliser_St Derived Data Type	582

Totaliser - Description

General

The Totaliser block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB Totaliser
Vijeo Citect	Genie Totaliser
UAG	SCoD Totaliser

Function Description

The Totaliser block is used for applications where totalise of value is required. An example is totalising production per day.

Initialise totalising command is given as input to the block, and it starts totalising the input value given as an input pin IN.

The divisor constant is given during initialisation. This constant is used to calculate the values that are related to time (for example, t/h, m/s). It must be time value of IN/TScan.

For example, if t/h is the input value, TScan = 1 s, then K = 3600. This constant will be used for parameters like tonnes/hour where hour is the divisor constant.

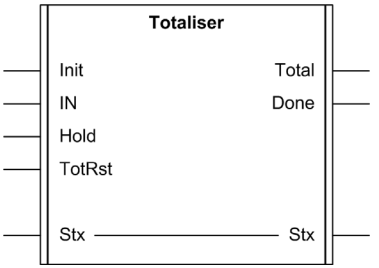
Hold command is given as an input to the block. If it is high, then calculation will be hold, and the last value will be retained.

Reset command is given as an input to the block. If it is high, then the output value (Total) will be reset to 0.

A set point is given to the block. If the total value is higher than the set point, the block gives an output (Done). This will be used in places where it has to trigger an event.

Representation in FBD

The following figure shows the FBD representation of the Totaliser block.



Parameter Description

The following table describes the input parameters of the Totaliser block:

Parameter	Type	Description	Details
Init	BOOL	Initialise totalising	The block starts to totalise when Init = 1 is received.
IN	REAL	Input	Input value to be totalised.
Hold	BOOL	Hold totalising	The block holds to totalising and holds the last Total output.
TotRst	BOOL	Reset	Total = 0 when TotRst = 1.

The following table describes the output parameters of the Totaliser block:

Parameter	Type	Description	Details
Total	REAL	Total value	Output value, Total = (In/K) + Total. Calculated during every TScan interval time. If TotRst = 1, then Total = 0. If Hold = 1, then Total value is not calculated.
Done	BOOL	Set point is reached	This output denotes that Total >= SP.

The following table describes the input/output parameter of the Totaliser block:

Parameter	Type	Description
Stx	Totaliser_St	Status and control of the analog actuator. Refer to the Structure to interface with the HMI *(see page 582)*.

`Totaliser` - Representation as SCoD in UAG

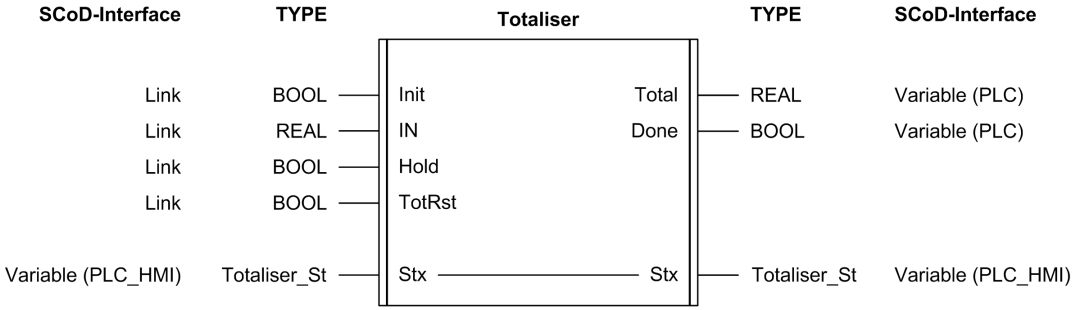
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the `Stx` variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type `HMI` is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the `Totaliser` block.



SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Display format (time)	`Stx.TScan`	#####	**Configuration**	1	Display format (time)
Measurement unit	`Stx.SP`	0.0	**Configuration**	2	Sets measurement unit
Time scan value (sec)	`Stx.TScan`	0	**Configuration**	3	Sets time scan value (s)
Display format (SP)	`Stx.SP`	0.0	**Configuration**	7	Display format (set point)
Set point	`Stx.SP`	0.0	**Configuration**	8	Sets set point
Divisor constant	`Stx.K`	0.0	**Configuration**	9	Divisor constant
Eng. Min	`Stx.SP`	0.0	**Range**	1	Sets minimum Eng. value
Eng. Max	`Stx.SP`	0.0	**Range**	7	Sets maximum Eng. value

`Totaliser` - Representation in the HMI

Totaliser (Genie)

The `Totaliser` block is represented in the HMI by the following genie.

Genie Elements

Symbol	Meaning
	Opens the `Totaliser` popup.
	Displays the total value of the totaliser.
	The instance name of the `Totaliser` block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

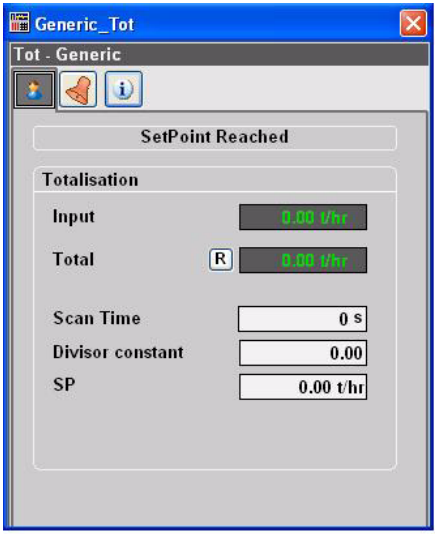
Item	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Popup

The `Totaliser` block popup has the following tabs:

- **Operator** tab
- **Alarm** tab
- **Information** tab

Operator Tab



This tab has 4 sub-sections:

- **Status**
- **Totalisation**

The sub-section **Status** displays the current status and the possible displays are listed as follows:

- **Stopped**
- **Initialize Totaliser**
- **Hold**
- **SetPoint Reached**

The following table describes the sub-section **Totalisation**:

Item	Meaning
Reset Total	Resets the total value to 0.
Input	Displays the input value to be totalised.
Total	Displays the totalised value.
TSCAN	Displays the scan time of the block to calculate.
Divisor Constant	Displays the divisor constant used to calculate.
SETPOINT	Displays the set point.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

Totaliser_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type Totaliser_St for the Totaliser block:

Element	Type	Description
K	REAL	Divisor constant
SP	REAL	Set point
IN	REAL	Input
Total	REAL	Totalised value
TScan	INT	Time scan value
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Dummy	WORD	Not used

Bit description for the Ctw element:

Bit	Description
0...6	Not used
7	Reset
8...15	Not used

Bit description for the Stw1 element:

Bit	Description
0	Initialize totalising
1	Hold totalising
2	Set point reached
3...15	Not used

Bit description for the Alw1 element:

Bit	Description
0...13	Not used
14	Total reached maximum value
15	Not used

11.6 EgySchE: Energy Scheduler Event Based

Overview

This section describes the EgySchE block.

What's in this Section?

This section contains the following topics:

Topic	Page
EgySchE - Description	584
EgySchE - Representation as SCoD in UAG	586
EgySchE - Representation in the HMI	587
EgySchE_St Derived Data Type	592

EgySchE - Description

General

The EgySchE block is implemented as listed in the following table:

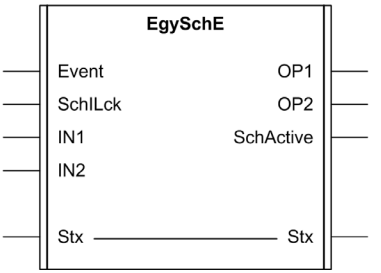
Software	Implemented as
Unity Pro	DFB EgySchE
Vijeo Citect	Genie EgySchE
UAG	SCoD EgySchE

Function Description

The EgySchE block is used for the applications where some actions need to be scheduled based on events, for example, shutting down of the mill fan. If mill is stopped for more than 30 minutes, the provided interlocks are satisfied. The process event logic is programmed and connected to the input pin Event. If Event is high, then schedule is activated. The schedule supports BOOL type and REAL type output. Normally, the input values are moved to the outputs. If the interlock (Schllck) and schedule are active, the default values are moved to outputs after a time delay SchDelay. The operator can enter these default values from the HMI popup.

If the interlock is not active after the time delay, the schedule further waits for a time (hold time). During initialization of the block, this time can be set and changed later. If the interlock becomes active within the hold time. The scheduler does its functions and shuts down itself, or else, the schedule gets cancelled and generates an alarm. In this, the operator gets information of remaining time, scheduler on/off status, and so on from the HMI popup.

Representation in FBD



Parameter Description

The following table describes the input parameters of the EgySchE block:

Parameter	Type	Description	Details
Event	BOOL	Event on	Schedule output is sent if Event is triggered.
Schllck	BOOL	Schedule interlock	Schedule block send output only if interlock = 1.
IN1	BOOL	Input1	Input 1
IN2	REAL	Input2	Input 2

The following table describes the output parameter of the EgySchE block:

Parameter	Type	Description	Details
OP1	BOOL	Output1	If SchActive = 1 and Schllck = 1, then OP1 = DV1, else OP1 = IN1.
OP2	REAL	Output2	If SchActive = 1 and Schllck = 1, then OP2 = DV2, else OP2 = IN2.
SchActive	BOOL	Schedule active	Schedule is active.

The following table describes the input/output parameter of the EgySchE block:

Parameter	Type	Description
Stx	EgySchE_St	Status and control of the EgySchE block. Refer to the Structure to interface with the HMI *(see page 592)*.

EgySchE - Representation as SCoD in UAG

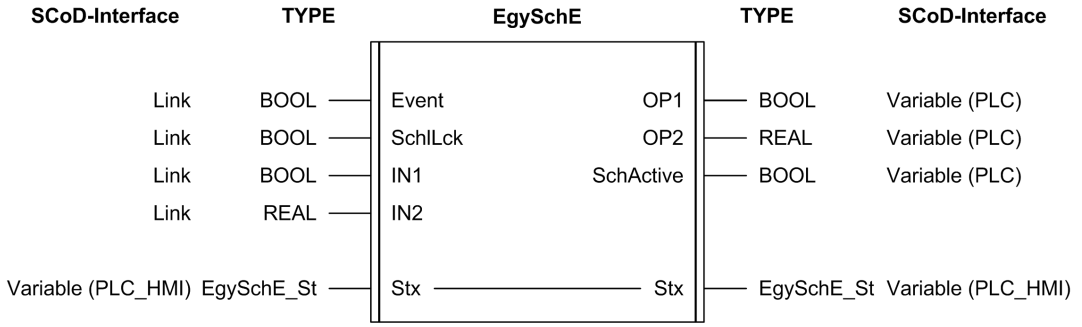
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the `Stx` variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type `HMI` is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the `EgySchE` block.



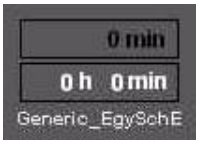
SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Default Value 1	`Stx.DV1`	0	**Configuration**	3	Default value 1
Default Value 2	`Stx.DV2`	0.0	**Configuration**	9	Default value 2
Display format	`Stx.OP2`	Not available	**Configuration**	1	Display format
Hold delay (min)	`Stx.THold`	0	**Configuration**	8	Hold time
Measurement Unit	`Stx.OP2`	Not available	**Configuration**	7	Measurement unit
Schedule delay (min)	`Stx.TSchedule`	0	**Configuration**	2	Scheduled time

EgySchE - Representation in the HMI

Energy Scheduler Event (Genie)

The EgySchE block is represented in the HMI by the following genie.



Genie Elements

Symbol	Meaning
	Opens the EgySchE popup.
	The instance name of the EgySchE block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

Item	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Animation Description of the Genie

Symbol	Animation	Meaning
	Line color: Flashing yellow	Alert alarm for the energy scheduler event based block is active.
	Line color: Steady yellow	Alert alarm for the energy scheduler event based block is acknowledged.

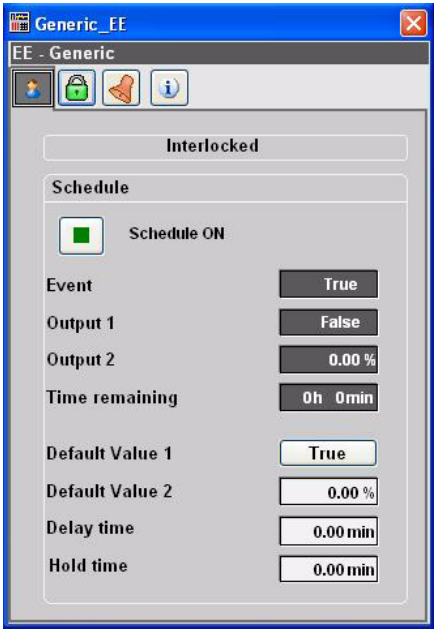
Symbol	Animation	Meaning
	Line color: Flashing red	Trip alarm for the energy scheduler event based block is active.
	Line color: Steady red	Trip alarm for the energy scheduler event based block is acknowledged.

Popup

The `EgySchE` block popup has the following tabs:

- **Operator** tab
- **Interlock** tab
- **Alarm** tab
- **Information** tab

Operator Tab



This tab has 2 sub-sections:

- **Status**
- **Schedule**

The sub-section **Status** displays the current statsus.

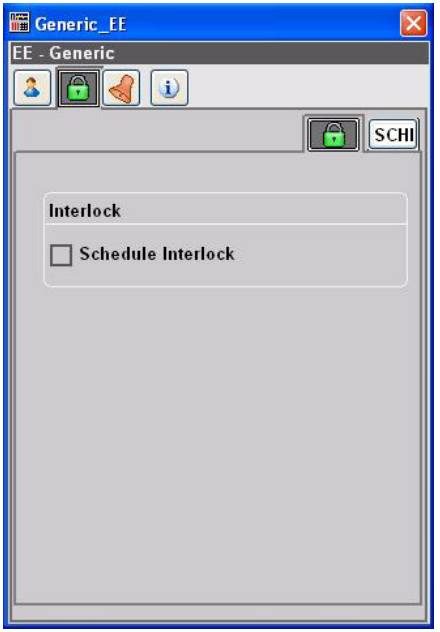
The possible displays are listed as follows:

- **Sleeping**
- **Schedule Active**
- **Hold**
- **Interlocked**
- **Warning**
- **Alarm**

The following table describes the sub-section **Schedule**:

Item	Meaning
Schedule ON	Operator can select the schedule to turn on or turn off.
Event	The event condition either true or false is displayed.
Output 1	The output condition either true or false is displayed.
Output 2	The output value of the block is displayed.
Time remaining	Time remaining for the schedule occurrence is displayed.
Default Value 1	You can select the default bool value, that is, true or false to be moved to **Output 1**.
Default Value 2	You can select the default real value to be moved to **Output 2**.
Delay time	You can configure the delay time.
Hold time	You can configure the hold time.

Interlock Tab



This tab has a sub-section called as **Interlock**, which is described in the following table:

Item	Animation	Description
Schedule Interlock	Steady green	Schedule interlock is active, and the relacted device is not blocked.
	Steady gray	Schedule interlock is not active, and the related device is blocked.

This tab has 2 sub-tabs, which are described in the following table:

Symbol	Meaning
	Navigates to main interlock page.
	If **SCHI** tab is selected, all scheduler interlock conditions are displayed with description. Also, individual interlock condition can be bypassed in this section.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

EgySchE_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type EgySchE_St for the EgySchE block:

Element	Type	Description
DV2	REAL	Default value 2
IN2	REAL	Input 2
OP2	REAL	Output 2
TSchedule	INT	Scheduled time
THold	INT	Hold time
TRemaining	INT	Remaining time
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI
DV1	BOOL	Default value 1
IN1	BOOL	Input 1
OP1	BOOL	Output 1
Dummy	BOOL	Not used

Bit description for the Ctw element:

Bit	Description
0	Schedule on
1...6	Not used
7	Reset
8...15	Not used

Bit description for the Stw1 element:

Bit	Description
0	Event on/in
1	Schedule interlock
2	Schedule active
3	Hold
4...15	Not used

Bit description for the `Alw1` element:

Bit	Description
0	Schedule inoperable
1	Schedule wait
2...15	Not used

11.7 EgySchT: Energy Scheduler Time Based

Overview

This section describes the EgySchT block.

What's in this Section?

This section contains the following topics:

Topic	Page
EgySchT - Description	595
EgySchT - Representation as SCoD in UAG	597
EgySchT - Representation in the HMI	598
EgySchT_St Derived Data Type	603

EgySchT - Description

General

The EgySchT block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB EgySchT
Vijeo Citect	Genie EgySchT
UAG	SCoD EgySchT

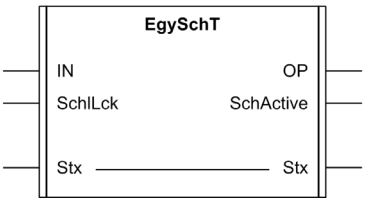
Function Description

The EgySchT block is used for the applications where some actions need to be scheduled based on time, for example, shutting down of feed extraction unit at night since cost of energy is high. The schedule is activated and the schedule time is entered by the operator from the HMI popup. The schedule supports BOOL and REAL type output. The input values are moved to output. If the interlock (Schllck) and schedule are active, and the controller clock time crosses the schedule time, the default values are moved to outputs. The operator enters the default values from the HMI popup.

If the interlock is not active when the schedule time is crossed, the scheduler further waits for a time (hold time). During initialization of the block, this time can be set and changed later. If the interlock becomes active within the hold time, the scheduler functions and shuts down. Else, the schedule is cancelled and an alarm is generated. The operator gets the information of remaining time, scheduler on/off status, and so on from the HMI popup.

Representation in FBD

The following figure shows the FBD representation of the EgySchT block.



Parameter Description

The following table describes the input parameters of the EgySchT block:

Parameter	Type	Description	Details
IN	BOOL	Input	Input
Schllck	BOOL	Schedule interlock	Schedule block is active only if interlock = 1.

The following table describes the output parameter of the EgySchT block:

Parameter	Type	Description	Details
OP	BOOL	Output	If SchActive = 1 and Schllck = 1, then OP = DV, else OP = IN.
SchActive	BOOL	Schedule active	Scheduler is active to take action.

The following table describes the input/output parameter of the EgySchT block:

Parameter	Type	Description
Stx	EgySchT_St	Status and control of the EgySchT block. Refer to the Structure to interface with the HMI *(see page 603)*.

EgySchT - Representation as SCoD in UAG

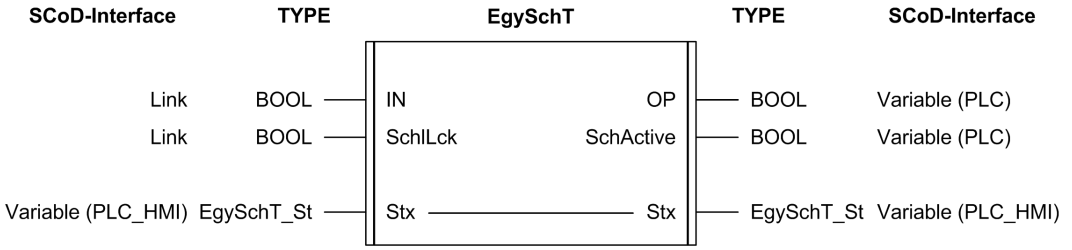
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the `Stx` variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type `HMI` is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the `EgySchT` block.



SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Default Value	`Stx.DV`	0	**Configuration**	7	Default value
Display format	`Stx.TSchedule`	0	**Configuration**	1	Display format
Hold delay (min)	`Stx.THold`	0	**Configuration**	8	Hold time
Schedule delay (min)	`Stx.TSchedule`	0	**Configuration**	2	Schedule delay (min)

EgySchT - Representation in the HMI

Energy Scheduler Time (Genie)

The EgySchT block is represented in the HMI by the following genie.

Genie Elements

Symbol	Meaning
	Opens the EgySchT popup. Enter the value of schedule time from the genie. To enter the value the operator should follow the format as (##h##m).
	The instance name of the EgySchT block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

Item	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Animation Description of the Genie

Symbol	Animation	Meaning
	Line color: Flashing yellow	Alert alarm for the energy scheduler time based block is active.
	Line color: Steady yellow	Alert alarm for the energy scheduler time based block is acknowledged.

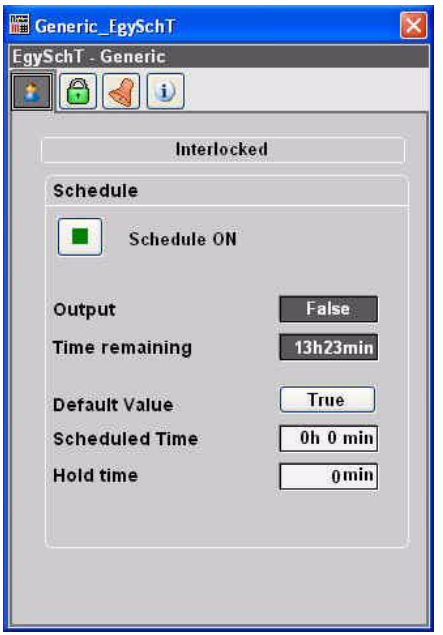
Symbol	Animation	Meaning
	Line color: Flashing red	Trip alarm for the energy scheduler time based block is active.
	Line color: Steady red	Trip alarm for the energy scheduler time based block is acknowledged.

Popup

The `EgySchT` block popup has the following tabs:

- **Operator** tab
- **Interlock** tab
- **Alarm** tab
- **Information** tab

Operator Tab



This tab has 2 sub-sections:

- **Status**
- **Schedule**

The sub-section **Status** displays the current status.

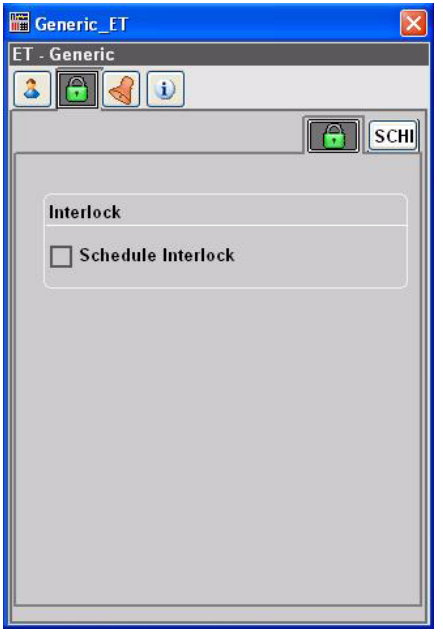
The possible displays are listed as follows:

- **Sleeping**
- **Schedule Active**
- **Schedule On**
- **Hold**
- **Interlocked**
- **Warning**
- **Alarm**

The following table describes the sub-section **Schedule**:

Item	Meaning
Schedule ON	The operator can select the schedule to turn on/turn off.
Repeat schedule ON	The operator can select the repeat on for the schedule.
Output	The output condition either true or false is displayed.
Time remaining	Time remaining for the schedule occurrence is displayed.
Default Value	You can select the default condition, that is, true or false.
Scheduled Time	You can configure the delay time.
Hold time	You can configure the hold time.

Interlock Tab



This tab has a sub-section called as **Interlock**, which is described in the following table:

Item	Animation	Description
Schedule Interlock	Steady green	Schedule interlock is active, and the related device is not blocked.
	Steady gray	Schedule interlock is not active and the related device is blocked.

This tab has 2 sub-tabs, which are described in the following table:

Symbol	Meaning
	Navigates to main interlock page.
	If **SCHI** tab is selected, all scheduler interlock conditions are displayed with description. Also, individual interlock condition can be bypassed in this section.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

EgySchT_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type EgySchT_St for the EgySchT block:

Element	Type	Description
TSchedule	INT	Scheduled time
THold	INT	Hold time
TRemaining	INT	Remaining time
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI
DV	BOOL	Default value
IN	BOOL	Input
OP	BOOL	Output
Dummy	BOOL	Not used

Bit description for the Ctw element:

Bit	Description
0	Schedule on
1...6	Not used
7	Reset
8 ... 15	Not used

Bit description for the Stw1 element:

Bit	Description
0	Event on/in
1	Schedule interlock
2	Schedule active
3	Hold
4...15	Not used

Bit description for the `Alw1` element:

Bit	Description
0	Schedule inoperable
1	Schedule wait
2 ... 15	Not used

11.8 ILCK: Interlock

Overview

This section describes the ILCK block.

What's in this Section?

This section contains the following topics:

Topic	Page
ILCK - Description	606
ILCK - Representation as SCoD in UAG	608
ILCK_St Derived Data Type	609

ILCK - Description

General

The ILCK block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB ILCK
Vijeo Citect	DFB ILCK
UAG	SCoD ILCK

Function Description

The ILCK block is used to provide a single interlock output for motors, valves, and other actuators that has multiple interlock conditions. This block will be connected in series before motors and actuator blocks. It represents any type of interlock (starting interlock or safety interlock or sequential interlock or process interlock) based on the application.

This block supports 8 interlock conditions. If all the Interlocks are active, the result of the interlock block is active. Hence the connected interlock pin in the motor or actuator is active. Individual bypassing of all the 8 interlock condition is available in the interlock block.

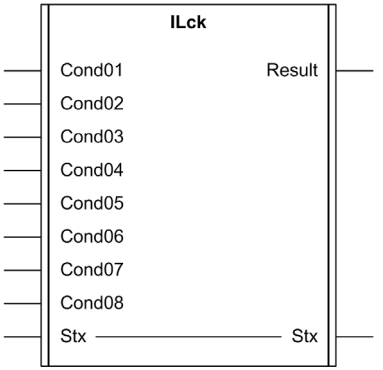
The detailed description of the individual interlock conditions with their status display and bypassing of the same will be made available in the **Interlock** tab of super genie of the corresponding motor or actuator.

NOTE: The requirement to view the description of the interlocks is that it must be named in the format: <Motor/Actuator name>_<Interlock type>

For example, if the name of the motor is KILN_FAN32, the interlock block name must be KILN_FAN32_STI or KILN_FAN32_SI or KILN_FAN32_SQI or KILN_FAN32_PI depending on its application.

Representation in FBD

The following figure shows the FBD representation of the ILCK block.



Parameter Description

The following table describes the input parameter of the ILCK block:

Parameter	Type	Description
Cond01	BOOL	Interlock condition 1
Cond02	BOOL	Interlock condition 2
Cond03	BOOL	Interlock condition 3
Cond04	BOOL	Interlock condition 4
Cond05	BOOL	Interlock condition 5
Cond06	BOOL	Interlock condition 6
Cond07	BOOL	Interlock condition 7
Cond08	BOOL	Interlock condition 8

The following table describes the output parameter of the ILCK block:

Parameter	Type	Description
Result	BOOL	Interlock is active

The following table describes the input/output parameter of the ILCK block:

Parameter	Type	Description
Stx	ILCK_St	Status and control of ILCK block. Refer to the Structure to interface with the HMI *(see page 609)*.

ILCK - Representation as SCoD in UAG

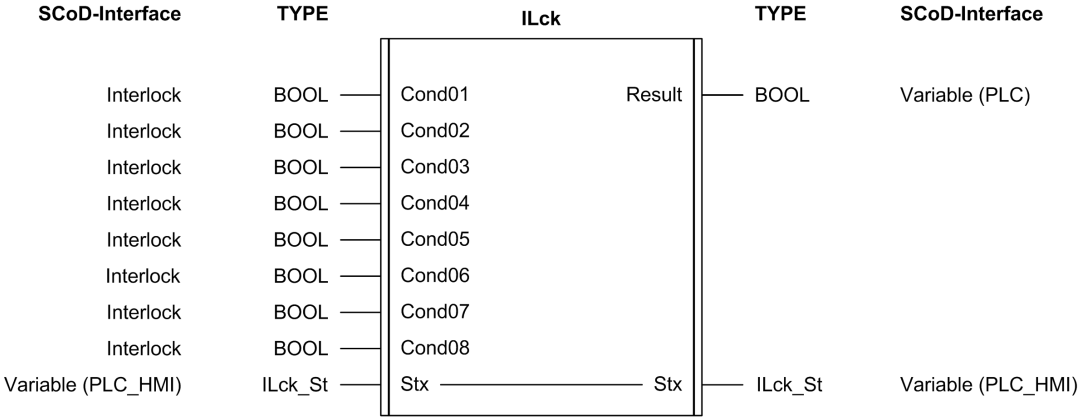
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the Stx variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type HMI is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following figure shows the input/output diagram of the SCoD-Interface for the ILCK block.



SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Condition1	ILCK.Cond01	Not available	**Configuration**	1	Condition 1
Condition2	ILCK.Cond02	Not available	**Configuration**	7	Condition 2
Condition3	ILCK.Cond03	Not available	**Configuration**	2	Condition 3
Condition4	ILCK.Cond04	Not available	**Configuration**	8	Condition 4
Condition5	ILCK.Cond05	Not available	**Configuration**	3	Condition 5
Condition6	ILCK.Cond06	Not available	**Configuration**	9	Condition 6
Condition7	ILCK.Cond07	Not available	**Configuration**	4	Condition 7
Condition8	ILCK.Cond08	Not available	**Configuration**	10	Condition 8

ILCK_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type ILCK_St for the ILCK block:

Element	Type	Description
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI

Bit description for the Ctw element:

Bit	Description
0...7	Not used
8...15	Bypass condition1...Bypass condition8

Bit description for the Stw element:

Bit	Description
0...6	Not used
7...14	Condition1...Condition8
15	Result

11.9 PIDB: Proportional, Integral and Derivative Controller Block

Overview

This section describes the PIDB block.

What's in this Section?

This section contains the following topics:

Topic	Page
PIDB - Description	611
PIDB - Representation as SCoD in UAG	619
PIDB - Representation in the HMI	621
PIDB_St Derived Data Type	629

PIDB - Description

General

The PIDB block is implemented as listed in the following table:

Software	Implemented as
Unity Pro	DFB PIDB
Vijeo Citect	Genie PIDB
UAG	SCoD PIDB

Function Description

This PIDB block is used to have a proportional, integral and derivative control over an actuator in order to maintain a process value close to given target.

This block supports 4 modes of operation:
- Auto mode
- Manual mode
- Cascade mode
- External mode

By default, the block will be in the mode defined by the input pin, DMode. The block will be forced to any one of the modes (Auto or Manual or Cascade or External modes) if the corresponding mode selection input pin (AutoSel or ManSel or CasSel or ExtSel) is HIGH. The block goes to Manual mode, if more than 1 pin is selected.

If the mode selection input pins (AutoSel or ManSel or CasSel or ExtSel) is LOW, then the mode of the PIDB block is changed by the operator from the popup of the block.

General Operation

A wake-up command (**Wake**) is given by the GrpCtrl block to wake-up the block from sleep.

Auto Mode Operation

In Auto mode, the operator sets the target or set point to be maintained (AutoSP). Based on this target and the process value, the detected error or deviation is derived.

The target is restricted within high and low limits (SPH and SPL), and if the operator enters the value beyond the limits, then the target value is equal to high or low limit.

Based on this deviation, the manipulating variable value (MV) is calculated using the PID algorithm and sent to the actuator to act.

The actuator action makes a process change and thereby the process value (PV) changes and the control continuously keeps calculating the control variable and the cycle goes on.

The MV value is always restricted within the high and low limits (MVH and MVL). If the calculation gives MV beyond the limits, then MV is equal to the value of high or low limit.

The maximum deviation (MaxError) allowed is configured. If the deviation (Error) exceeds the maximum deviation value for longer time (TMaxError), then an alarm is generated.

The minimum deviation (MinError) for control is configured. If the deviation (Error) is less than the minimum deviation value for longer time (TMinError), then the control algorithm is bypassed and the manipulated value is holded at its last value.

The maximum step change (Slope) for manipulating variable (MV) is configured. If difference of present MV and previous MV is more than slope, then MV = PrevMV + Slope.

Manual Mode Operation

In Manual mode, the operator sets the set point (ManSP) to the control variable (MV) directly from the popup.

The operator can enter set point only within the high and low limits (MVH and MVL). If the operator enters the value beyond the limit, then the control variable value is equal to high or low limit.

Cascade Mode Operation

In Cascade mode, the set point to the control variable is given as an input to the block (CasSP). The target is restricted within high and low limits (SPH and SPL), and if the input value (CasSP) is beyond the limits, then the target value is equal to high or low limit.

Based on this target and the process value, the detected error or deviation is derived. Based on this error, the manipulating variable value (MV) is calculated using the PID algorithm and sent to the actuator to act.

The actuator action makes a process change and thereby the process value (PV) changes and the manipulating variable continuously keeps varying based on the calculation and the cycle goes on.

The MV value is always restricted within the high and low limits (MVH and MVL). If the calculation gives MV beyond the limits, then MV is restricted to the value of high or low limit.

The maximum deviation, minimum deviation and maximum step change concepts are applicable as in Auto mode.

External Mode Operation

In External mode, the manipulating variable is given as an input to the block (ExtSP) directly from the popup.

If input value (ExtSP) is beyond the limit (MVH and MVL), then the value is equal to high or low limit.

Configuration

If auto set point tracking is enabled, then auto set point value is tracked to the current target SP (AutoSP = SP). This is mainly done to maintain bump less transfer during mode change from Cascade to Auto mode.

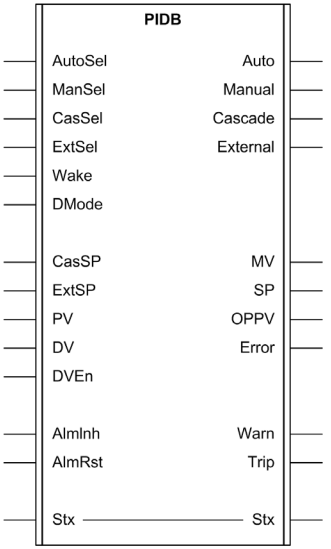
If manual set point tracking is enabled, then manual set point value is tracked to the current output (MV) (ManSP = MV). This is mainly done to maintain bump less transfer during mode change from modes (Auto or Cascade or External) to Manual mode.

If process value tracking is enabled, then process value is tracked to the current output MV (PV = MV). This is mainly done to hold MV at the same value when the block is in Auto or Cascade mode. If the default value is enabled and given as an input to the block, it is moved to the output during initialisation.

The alert and trip is indicated in the SCADA. The details of the active alarms are shown in the **Alarm** tab of the popup. Based on user configured conditions, the alarms for this block can be inhibited by connecting the logic to the input pin (AlmInh). The trip alarms can be reset by the input pin (AlmRst) that is connected from GrpCtrl block or by the operator from popup.

Representation in FBD

The following figure shows the FBD representation of the `PIDB` block.



Parameter Description

The following table describes the input parameters of the `PIDB` block:

Parameter	Type	Description	Details
`AutoSel`	`BOOL`	Auto mode	Selection of mode of operation (Auto mode): ● `AutoSel` = 1: Block is forced to Auto mode. Mode Selection option in HMI is grayed out. ● `AutoSel` = 0: Block is set free for operator to select the mode from HMI.
`ManSel`	`BOOL`	Manual mode	Selection of mode of operation (Manual mode): ● `ManSel` = 1: Block is forced to Manual mode. Mode selection option in HMI is grayed out. ● `ManSel` = 0: Block is set free for operator to select the mode from HMI.

Parameter	Type	Description	Details
CasSel	BOOL	Cascade mode	Selection of mode of operation (Cascade mode): ● CasSel = 1: Block is forced to Cascade mode. Mode selection option in HMI is grayed out. ● CasSel = 0: Block is set free for operator to select the mode from HMI.
ExtSel	BOOL	External mode	Selection of mode of operation (External mode): ● ExtSel = 1: Block is forced to Auto mode. Mode selection option in HMI is grayed out. ● ExtSel = 0: Block is set free for operator to select the mode from HMI.
Wake	BOOL	Group wake-up	Group wake-up and request for group availibility check: ● Wake = 1: Group is active. ● Wake = 0: Group is sleeping. This input is connected from output GWake of GrpCtrl.
DMode	INT	Default mode	This Input defines the default mode of the block when it is initialized: ● 1 : Auto ● 2 : Manual ● 3 : Cascade ● 4 : External
CasSP	REAL	Cascade mode set point	If the block is in Cascade mode, CasSP is moved to SP and MV is calculated as per the SP.
ExtSP	REAL	External mode set point	If the block is in External mode, ExtSP is moved to MV.
PV	REAL	Start-up value enable	Process value
DV	REAL	Start-up value	If DVEn = 1, then this value is moved to output MV during initialization.
DVEn	BOOL	Start-up value enable	When DVEn = 1, the DV is moved to output MV during initialization.

⚠ CAUTION
UNINTENDED MACHINE OR PROCESS BEHAVIOR Be sure of the state of security of your machine or process environment. The `AlmInh` parameter to inhibit alarms must be used very carefully. **Failure to follow these instructions can result in injury or equipment damage.**

The following table describes the input parameter of the `PIDB` block:

Parameter	Type	Description	Details
`AlmInh`	`BOOL`	Alarm inhibition	• `AlmInh` = 1: Except for safety alarms, all the other alarms are inhibited. No alarms are generated by the block. • `AlmInh` = 0: Alarms are not inhibited.

⚠ CAUTION
UNINTENDED MACHINE OR PROCESS BEHAVIOR Be sure of the state of security of your machine or process environment. To reset, all generated alarms must be used very carefully. **Failure to follow these instructions can result in injury or equipment damage.**

The following table describes the input parameter of the `PIDB` block:

Parameter	Type	Description	Details
`AlmRst`	`BOOL`	Reset	• `AlmRst` = 1: Resets all the alarms generated by the block. • `AlmRst` = 0: Not resets all the alarms generated by the block. This is generally a pulse input. This must be connected from `GRst` pin of `GrpCtrl` block and resets the trip alarm.

The following table describes the output parameters of the PIDB block:

Parameter	Type	Description	Details
Auto	BOOL	Auto mode	The block is in Auto mode. Output MV is generated based on SP. SP = ManSP and is set by the operator from HMI.
Manual	BOOL	Manual mode	The block is in Manual mode. MV = ManSP and is set by the operator from HMI.
Cascade	BOOL	Cascade mode	The block is in Cascade mode. Output MV is generated based on CasSP.
External	BOOL	External mode	The block is in External mode. MV = ExtSP and input pin is connected from other blocks or systems.
MV	REAL	Manipulated value	This is the final output that is sent as the set points to acuators (for example, Valve position, Fan speed,and so on).
SP	REAL	Set point	This is the final set point based on which MV is generated. Applicable only in Auto and Cascade modes of operation.
OPPV	REAL	Output process value	This is the final process value based on which error is calculated.
Error	REAL	Error	This value is the deviation of the process value from target. Error = SP - PV.
Warn	BOOL	Alert	• Warn = 1: The control is operational with no active alert present. • Warn = 0: The control is not operational with an active alert present.
Trip	BOOL	Trip	• Trip = 1: The control is operational with no active trip alarm present. • Trip = 0: The control is not operational with an active alarm present.

The following table describes the input/output parameter of the PIDB block:

Parameter	Type	Description
Stx	PIDB_St	Status and control of the PIDB block. Refer to the Structure to interface with the HMI *(see page 629)*.

PIDB - Representation as SCoD in UAG

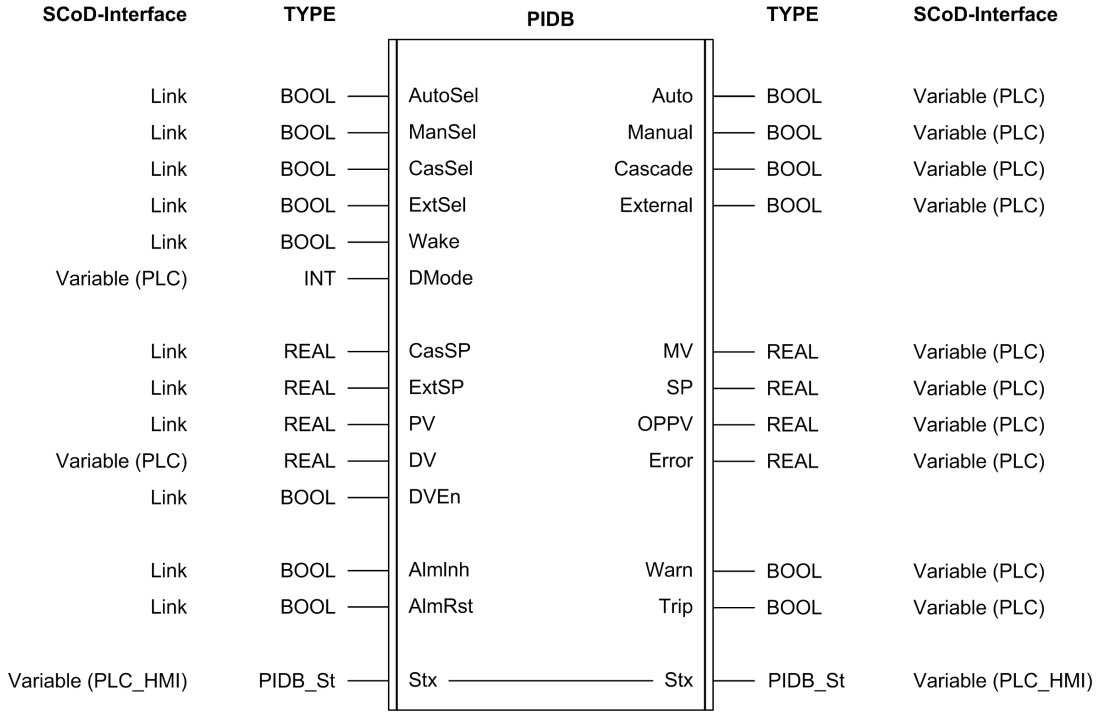
Overview

The use of DFB pins in the SCoD editor is explained in the input/output diagram. The variables defined in the SCoD editor have the same name as the pin. The update initial value is set for the members (child variables) of the `Stx` variable in the UAG. The values can be modified and updated manually in Unity Pro before the PLC is restarted.

Additional variable of type `HMI` is declared without a connection to the pin. This variable holds the instance name of the function block for the HMI.

Input Output Diagram

The following diagram shows the input/output diagram of the SCoD-Interface for the `PIDB` block.



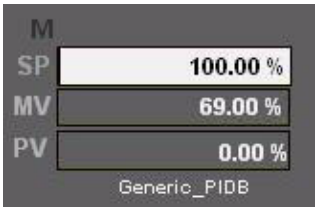
SCoD Properties

Name	Variable	Initial Value	Property Tab	Position	Description
Display format (time)	Stx.TInt	#####	**Configuration**	1	Display format (time)
Display format (MV)	Stx.MV	#####	**Configuration**	2	Display format (Manipulating Variable)
Measurement Unit (MV)	Stx.MV	Not available	**Configuration**	3	Measurement unit (Manipulating Variable)
Default Value	DV	0.0	**Configuration**	4	Default value
Eng. Max (MV)	Stx.MV	Not available	**Configuration**	5	Sets maximum Eng. value (Manipulating Variable)
Display format (SP)	Stx.SP	###.##	**Configuration**	7	Display format (Set point)
Measurement Unit (SP)	Stx.SP	Not available	**Configuration**	8	Measurement unit
Default Mode	DMode	1-Auto	**Configuration**	9	Default mode
Set Point Low	Stx.SPL	0.0	**Limit**	1	Set point low
Set Low MV	Stx.MVL	0.0	**Limit**	2	Manipulating variable low limit
Set Minimum Error	Stx.MinError	0.0	**Limit**	3	Sets minimum error
Set Minimum Errror Time (sec)	Stx.TMinError	0	**Limit**	4	Sets minimum errror time
Set Point High	Stx.SPH	100.0	**Limit**	7	Set point high
Set High MV	Stx.MVH	100.0	**Limit**	8	Manipulating variable high limit
Set Maximum Error	Stx.MaxError	0.0	**Limit**	9	Sets maximum error
Set Maximum Error Time (sec)	Stx.TMaxError	0	**Limit**	10	Sets maximum error time
Eng. Min (SP)	Stx.SP	Not available	**Range**	1	Sets minimum Eng. value (Set point)
Eng. Min (MV)	Stx.MV	Not available	**Range**	2	Sets minimum Eng. value (Manipulating Variable)
Eng. Max (SP)	Stx.SP	Not available	**Range**	7	Sets maximum Eng. value (set point)

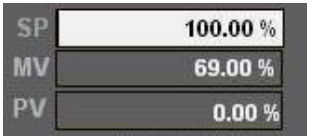
PIDB - Representation in the HMI

Proportional, Integral and Derivative Basic (Genie)

The PIDB block is represented in the HMI by the following genie.

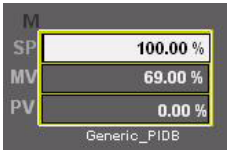
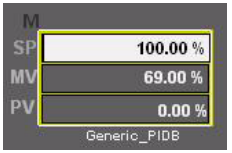
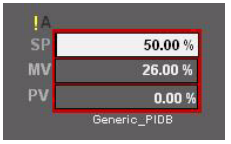
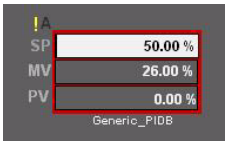


Genie Elements

Symbol	Meaning
	Opens the PIDB popup.
	Displays the set point value of the PIDB block. The operator can enter the value through genie.
	Displays the manipulating value of the PIDB block.
	Displays the process value of the PIDB block.
	The instance name of the PIDB block is visible by default. You can hide/unhide it by clicking the **Hide**/**Unhide** button in the template.

Item	Meaning
!	Exclamation mark is displayed in yellow color on the left corner if operator left any notes in the **Information** tab of the popup.

Animation Description of the Genie

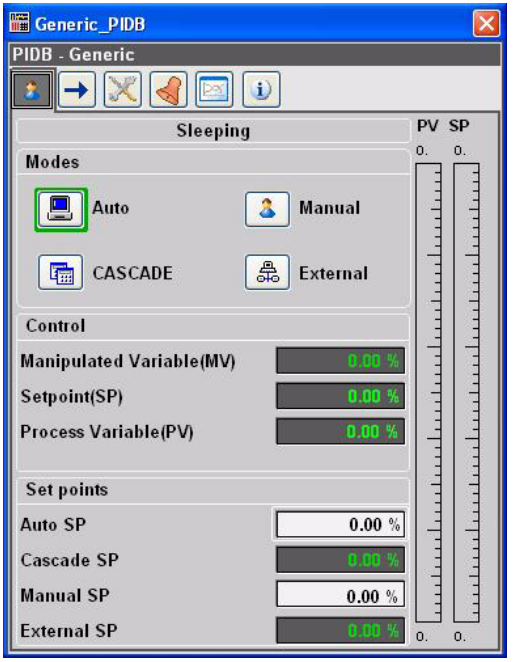
Symbol	Animation	Meaning
	Line color: Flashing yellow	Alert alarm for the energy `PIDB` block is active.
	Line color: Steady yellow	Alert alarm for the `PIDB` block is acknowledged.
	Line color: Flashing red	Alarm for the energy `PIDB` block is active.
	Line color: Steady red	Alarm for the `PIDB` block is acknowledged.

Popup

The `PIDB` block popup has the following tabs:

- **Operator** tab
- **Extended** tab
- **Configuration** tab
- **Alarm** tab
- **Trend** tab
- **Information** tab

Operator Tab



This tab displays the status of the `PIDB` block operation. This tab also displays the bar graph of set point and process variable.

This tab has 3 sub-sections:

- **Status**
- **Modes**
- **Control**
- **Set points**

The sub-section **Status** displays the current status.

The possible displays are listed as follows:

- **Sleeping**
- **Awake**
- **Default value Enabled**
- **Warning**
- **Alarm**

The following table describes the sub-section **Modes**:

Item	Description
Auto	The manipulating variable is generated based on the PID algorithm with reference to the set point entered by the operator.
Manual	The manipulating variable is replaced with the value entered by the operator from popup.
CASCADE	The manipulating variable is generated based on the PID algorithm with reference to the set point connected through the input pin.
External	The manipulating variable is replaced with the value coming from the input pin.

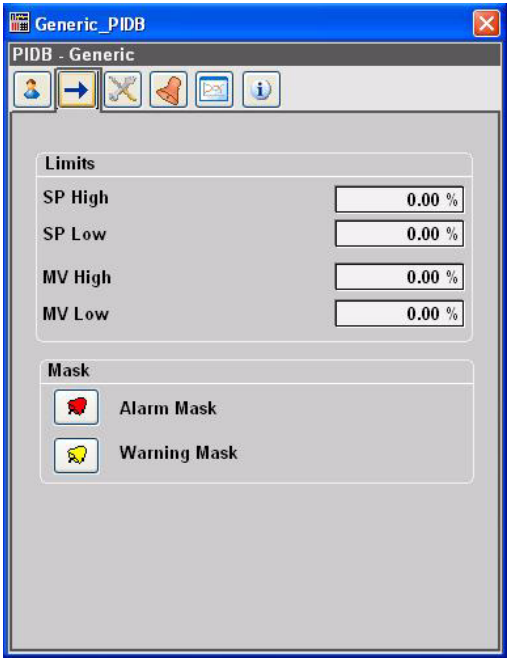
The following table describes the sub-section **Control**:

Item	Description
Manipulated Variable(MV)	Manipulated variable is the final output of the PID that is varied as per the detected error and given to an actuator to reduce the detected error. For example, speed - 100 RPM, position - 55 %, and so on.
Setpoint(SP)	The set point is the target value from the process parameter, that is, intended to be achieved by the PID by varying the `MV`. For example, temperature - 500 °c, pressure - 100 bar, and so on.
Process variable(PV)	The actual value feedback of the process parameter which is intended to be controlled by the `PIDB` block.

The following table describes the sub-section **Set points**:

Item	Description
Auto SP	Auto mode set point. This value is entered by the operator from HMI and is moved to `SP`.
Cascade SP	Cascade mode set point. This value is connected from any other block as input pin and is moved to `SP`.
Manual SP	Manual mode set point. This value is entered by the operator from HMI and is moved to `MV`.
External SP	External mode set point. This value is connected from any other block or system (for example, advanced process control system) and is moved to `MV`.

Extended Tab



This tab has 2 sub-sections:

- **Limits**
- **Mask**

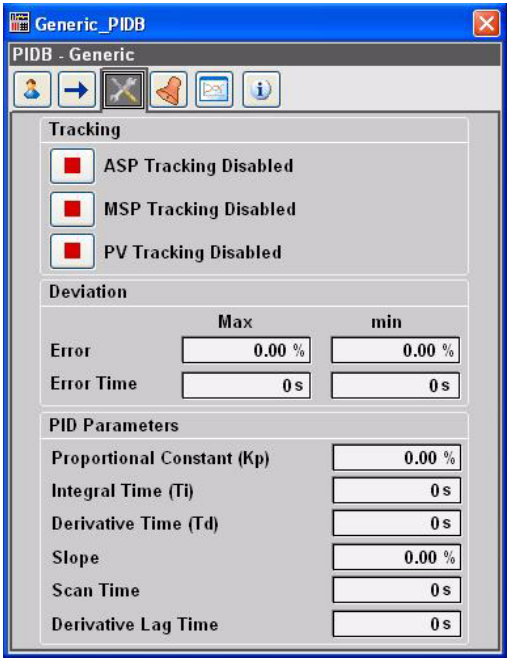
The following table describes the sub-section **Limits**:

Item	Description
SP High	Set point high limit.
SP Low	Set point low limit.
MV High	Manipulating variable high limit.
MV Low	Manipulating variable low limit.

The following table describes the sub-section **Mask**:

Item	Description
Alarm Mask	Masks the current alarms of this block in the alarm list.
Warning Mask	Masks the current alerts of this block in the alarm list.

Configuration Tab



This tab has 3 sub-sections:

- **Tracking**
- **Deviation**
- **PID Parameters**

The following table describes the sub-section **Tracking**:

Item	Description
ASP Tracking Disabled	Enables the auto set point tracking when in Cascade mode.
MSP Tracking Disabled	Enables the manual set point tracking when in Auto/Cascade/External mode.
PV Tracking Disabled	Enables the process value tracking in all the modes thereby making the process value equal to the `SP`.

The following table describes the sub-section **Deviation**:

Item	Description
Max Error	Displays the maximum deviation permissible to be configured.
Min Error	Displays the minimum deviation permissible to be configured.
Max Error Time	Displays the maximum error time after which an alarm is generated to be configured.
Min Error Time	Displays the minimum error time after which the control variable is hold at its last value to be configured.

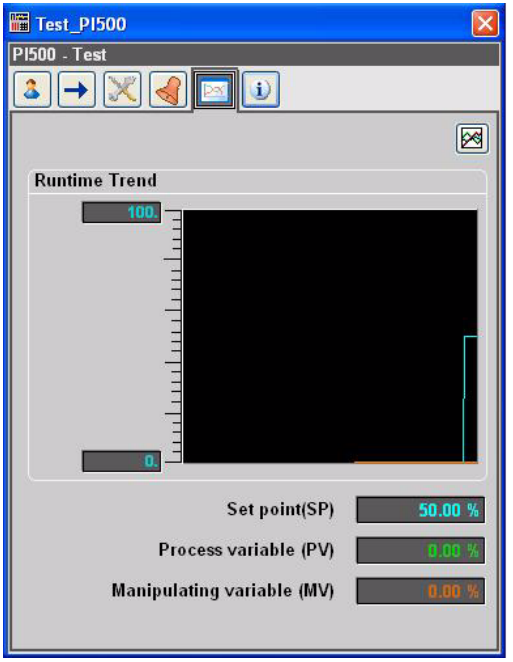
The following table describes the sub-section **PID Paramters**:

Item	Description
Proportional Constant (Kp)	Displays the proportional constant `Kp`.
Integral Time (Ti)	Displays the integral time `Ti`.
Derivative Time (Td)	Displays the derivative time `Td`.
Slope	Displays the slope.
Scan Time	Displays the scan time.
Derivative Lag Time	Displays the derivative lag time.

Alarm Tab

For description about **Alarm** tab, refer to Commonly Used Tabs *(see page 51)*.

Trend Tab



The sub-section **Runtime Trend** displays the run time trend of the parameters in the control module.

Item	Description
Set point(SP)	Displays run time trend of set point.
Process variable (PV)	Displays run time trend of process variable.
Manipulating variable (MV)	Displays run time trend of manipulating variable.
	Displays the historical trend page.

Information Tab

For description about **Information** tab, refer to Commonly Used Tabs *(see page 52)*.

PIDB_St Derived Data Type

Structure Description

The following table describes the structure of the derived data type PIDB_St for the PIDB block:

Element	Type	Description
AutoSP	REAL	Auto set point
CasSP	REAL	Cascade set point
SP	REAL	Set point
SPH	REAL	Set point high limit
SPL	REAL	Set point low limit
OPPV	REAL	Process value
Error	REAL	Error
MaxError	REAL	Maximum error
MinError	REAL	Minimum error
ManSP	REAL	Manual set point
ExtSP	REAL	External set point
MV	REAL	Manipulating variable
MVH	REAL	Manipulating variable high limit
MVL	REAL	Manipulating variable low limit
Slope	REAL	Slope
Kp	REAL	Proportional constant
TInt	INT	Integration time
TDeriv	INT	Derivative time
TDerivLag	INT	Derivative time lag
TMaxError	INT	Maximum error time
TMinError	INT	Minimum error time
TScan	INT	Scan time
Ctw	WORD	Control word to interact with HMI
Stw1	WORD	Status word 1 to interact with HMI
Alw1	WORD	Alarm word 1 to interact with HMI
Dummy1	WORD	Not used

Bit description for the `Ctw` element:

Bit	Description
0	Auto mode selection
1	Manual mode selection
2	Cascade mode selection
3	External mode selection
4	Auto set point track enable
5	Manual set point track enable
6	Process variable track enable
7	Alarm reset
8...13	Not used
14	Alarm mask
15	Alert mask

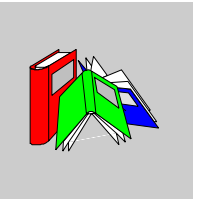
Bit description for the `Stw1` element:

Bit	Description
0	Wake
1	Auto selection mode
2	Manual selection mode
3	Cascade selection mode
4	External selection mode
5	Auto mode
6	Manual mode
7	Cascade mode
8	External mode
9	Default value enable
10...12	Not used
13	Alert
13	Alarm inhibition
15	Trip

Bit description for the `Alw` element:

Bit	Description
0	Detected error is high
1...2	Not used
3	More than 1 mode is selected in input pin
4...15	Not used

Glossary



A

Alm

alarm

Alw

alarm word

Anlg

analog

APV

analog processed value

ATV

Altivar

Auto

automatic

Avail

available

B

BES

belt empty sensor

BSS

belt sway switch

BTS

belt tension switch

C

C

calculation

Calv

calculation value

CAN

Controller Area Network

Cas

cascade

CFGW

configuration word

CHIN

channel input

CHOUT

calculated value of the output

CJS

chute jam switch

Cond

interlcok condition

ConvK

conversion constant

Ctrl

control

Ctw

control word

Cyc

cycle

D

Deriv

derivative

Dev

deviation

DFB

derived function block

Dig

digital

DP

differential pressure

DV

default value

E

En

enable

Ess

essential

EStop

emergency stop

EU

Engineering Unit: You can select it from the available list, configured as measurement unit groups in SCoD.

Ext

external

F

Fb

feedback

Fwd

forward

G

G, Grp

group

H

H

high

HH

high high

HMI

human machine interface

Hop

hopper

Hr

hours

Hyst

hysteresis

I

Ilck

interlock

IN

input

Inh

inhibition

Init

initialize

Inv

inverse

K

K

constant

Kp

proportional constant

L

L

low

LL

low low

Lvl

level

M

Man

manual

Max

maximum

Meas

measurement

Min

minimum

MV

manipulated value

O

Oos

out of service mode

OP

output

operator screen

The operator screen is a screen in Unity Pro.

P

PCN

pull card switch number

PCS

pull card switch

PID

proportional integral differential

PL

pulse length

popup

The popup window will be opened from genies.

Pos

position

Press, Pr

pressure

Proc

process

Pul

pulse gap time

PV

process value

R

Rev

reverse

rpm

revolutions per minute

Rst

alarm reset

S

SCADA

Supervisory Control and Data Acquisition

Sch

schedule

SCoD

Smart Control Device: A SCoD is a control module type defining the specific functionality of a control module.

Sel

selection

Seq

sequential

Sfty

safety

Sim

simulation

SP

set point

Stw

status word

Sw

switch

T

T

time

t/h

tons per hour

template

Template is a background screen (in Vijeo Citect) where genies and their popups get open.

U

UAG

Unity Application Generator

V

Val

value

W

Wake

wake-up

Warn

warning

Index

C

Cement Process
- GasAnalyser: Gas Analyser, *462*
- GirthGearLubCtrl: Girth Gear Lubrication Control, *512*
- MultiFuelCtrl: Multi Fuel Control, *487*

E

Equipment Control
- AnlgAct: Analog Actuator, *287*
- AnlgIn: Analog Input, *161*
- AnlgOut: Analog Output, *174*
- DigAct: Digital Actuator, *236*
- DigIn: Digital Input, *185*
- DigOut: Digital Output, *198*
- Mot1d: Motor 1 Direction, *106*
- Mot1dVS: Motor 1 Direction Variable Speed, *207*
- Mot2d: Motor 2 Direction, *133*
- MotAct: Motor Actuator, *260*
- MotBasic: Motor Basic, *316*

G

General Information, *17*
- Basics of Cement Library, *19*
- Function Blocks, *37*
- Functional Description, *25*
- Implementation Paradigms, *77*
- Instantiation in UAG, *91*
- Representation in the HMI, *41*
- Using the Cement Library, *55*

Generic Process
- EgySchE: Energy Scheduler Event Based, *583*
- EgySchT: Energy Scheduler Time Based, *594*
- ILCK: Interlock, *605*
- PIDB: Proportional, Integral and Derivative Controller Block, *610*
- RatioCtrl: Ratio Controller, *555*
- SPBool: Set Point Bool, *533*
- SPInt: Set Point Integer, *539*
- SPReal: Set Point Real, *547*
- Totaliser: Totaliser, *575*

M

MMM Process
- BagFltr: Bag Filter Sequence Block, *439*
- BCnvyr1d: Belt Conveyor 1 Direction, *385*
- GrpCtrl: Group Control, *336*
- GrpStpMgt: Group Stop Management,

371
GrpStrtMgt: Group Start Management, *357*
RouteMgt: Route Management, *418*

O

Object Description, *103*
Cement Process, *461*
Equipment Control, *105*
Generic Process, *531*
MMM Process, *335*

U

Using the Cement Library
Access Level Support, *60*
Acknowledge Alarms, *65*
Alarm Logging and Alarm Summary, *57*
Alarms in the Popup, *59*
Color Customization, *70*
Dim/Unavailable Buttons, *58*
Enable Use of Double Monitor, *62*
Information Popup, *74*
Local Language Support, *66*
Navigation Buttons, *73*
Navigation to Trend and Alarm Window from the Popup, *68*
Operator Logging, *56*
Out of Service Mode, *75*
Trend Assignment, *69*